Sequenzoptimierung eines synthetischen bifunktionalen Proteins durch multikriterielle genetische Algorithmen

Inaugural-Dissertation

zur Erlangung des Doktorgrades
Dr. rer. nat.

der Fakultät für Biologie an der
Universität Duisburg-Essen

vorgelegt von
Jonas Winkler
aus Münster

Essen,

Oktober 2012

Die der vorliegenden Arbeit zugrunde liegenden Experimente wurden am Zentrum für Medizinische Biotechnologie (ZMB) in der Abteilung für Bioinformatik der Universität Duisburg-Essen durchgeführt.

1. Gutachter: Prof. Dr. Daniel Hoffmann
2. Gutachter: PD Dr. Dominik Heider
3. Gutachter: Prof. Dr. Sven Rahmann

Vorsitzende des Prüfungsausschusses: Prof. Dr. Andrea Vortkamp

Tag der mündlichen Prüfung: 19. April 2013

«Some of the best things in life will come to you when you're not looking for them.»

I. Nath

Vor 25 Jahren begegnet, vor 15 Jahren wieder getroffen, seit 12 Jahren verliebt.

Wenn ich auch nicht schon damals danach gesucht habe: Ich will mein Leben mit dir verbringen.

Das ist für dich!

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Abkürzungsverzeichnis

Abb	Abbildung
Abs	Absatz
APBS	Adaptive Poisson-Boltzmann Solver
BCP	Population aus den besten Individuen (engl: *best chromosome population*)
bHLH	Helix-loop-helix-Transkriptionsfaktoren (engl: *basic-helix-loop-helix*)
BLAST	Basic Local Alignment Search Tool
CCP	Population aus den Kind-Individuen (engl: *child chromosome population*)
DNA	Desoxyribonukleinsäure (engl: *deoxyribonucleic acid*)
Fab	Fab Region eines Antikörpers, zuständig für die Antigenbindung
Fc	Fc Region eines Antikörpers, zuständig für die Immunantwort
GA	Genetischer Algorithmus
GROMACS	Groningen Machine for Chemical Simulations
H-Brücke	Wasserstoffbrückenbindung
HV	Hypervolumen-Indikator
IgG	Immunglobulin G
LSSC	Stoppkriterium der kleinsten Quadrate (engl: *least squares stopping criterion*)
MD	Molekülardynamik
MDR	Indikator der gegenseitigen Dominanzrate (engl: *mutual domination rate*)
MyoD	Myogener Faktor 3 (engl: *myoblast determination protein*)
NMR	Kernspinresonanz (engl: *nuclear magnetic resonance*)
PCP	Population proportional zur Fitness (engl: *proportional chromosome population*)
PDB	Protein Data Bank
RMSD	mittlere quadratische Abweichung (engl: *root-mean-square deviation*)

RMSF	mittlere quadratische Fluktuation (engl: *root-mean-square fluctuation*)
SAS	Von der Lösung erreichbare Fläche (engl: *solvent accessible surface*)
SpA	Staphylococcus Protein A
Tab	Tabelle

Aminosäuren:

Ala, A	Alanin
Arg, R	Arginin
Asn, N	Asparagin
Asp, D	Asparaginsäure
Cys, C	Cystein
Gln, Q	Glutamin
Glu, E	Glutaminsäure
Gly, G	Glycin
His ,H	Histidin
Ile, I	Isoleucin
Leu, L	Leucin
Lys, K	Lysin
Met, M	Methionin
Phe, F	Phenylalanin
Pro, P	Prolin
Ser, S	Serin
Thr, T	Threonin
Trp, W	Tryptophan
Try, Y	Tyrosin
Val, V	Valin

1 Einleitung

«Things don't have to change the world to be important.»

Steve Jobs

1.1 Prolog

Computer sind in der Biologie zu einem unentbehrlichen Hilfsmittel der täglichen Arbeit geworden. In vielen Aufgabenbereichen liefern sie die nötige Rechenleistung, um schnell verlässliche Ergebnisse zu ermitteln. So wären etwa die großen Genom-Sequenzierungsprojekte [52] oder verschiedene Klassifizierungsaufgaben[24, 38] ohne Computer nicht zu bewältigen gewesen. Mit der Zeit hat sich aus der Biologie und der Informatik das Feld der Bioinformatik gebildet, welches sich mit diesen komplexen Algorithmen und Programmen beschäftigt.

Das computergestützte Design von Proteinen bildet schon lange ein wichtiges Feld in der bioinformatischen Forschung. Laufend finden sich in der Literatur neue Designalgorithmen [68]. Viele Ansätze basieren auf heuristischen oder *trial-and-error* Methoden unter Zuhilfenahme von biologischem Wissen. Hierunter fallen etwa Faltungsalgorithmen oder gerichtete Evolutionssimulation. Mit diesen Algorithmen lassen sich Proteinsequenzen finden, die in eine gegebene Struktur falten.

Empirische Studien konnten zeigen, dass ähnliche Sequenzen in ähnliche oder gar gleiche Strukturen falten [2]. Jedoch gibt es auch Beispiele für wenig homologe Sequenzen mit beinahe gleicher Faltung [18, 48] wie die Antifrost-Glycoproteine [10], Protein-Prosphatasen [61] und Glytaminyl-Cyclasen [72]. Dies ist möglich, da auch

unterschiedliche Sequenzen ähnliche oder gleiche physikalische und chemische Eigenschaften besitzen können. In beiden Fällen gilt jedoch, dass die Struktur eines Proteins dessen Funktion vorgibt. Diese Tatsache soll in dieser Arbeit genutzt werden, um Sequenzen zu finden, die in eine spezifische Struktur falten und damit eine definierte Funktion erfüllen.

Ziel dieser Arbeit ist es nicht, alle möglichen Sequenzen zu finden, die in die gleiche Struktur falten. Vielmehr soll eine Lösung gefunden werden, die vorher definierte Bedingungen möglichst gut erfüllt: etwa die Stabilität, das Hydrophobizitätsprofil, die Elektrostatik oder eine katalytische Aktivität [81]. Es finden sich zahlreiche Beispiele für die erfolgreiche Anwendung des Proteindesigns, zum Beispiel Biokatalysatoren [46, 70, 93], Proteine mit verbesserter Bindeaffinität [51, 76] oder gesteigerter Stabilität [34].

Es hat sich gezeigt, dass das Design von Proteinen mit gegebener Funktion weitaus schwieriger ist, als nur die Modellierung der Proteinstruktur [30]. Besonders das Modellieren bi- oder multifunktionaler Proteine stellt eine große Herausforderung dar. Ferner ist der Zeitaufwand für den Designprozess auf Grund von komplexen Bewertungsfunktionen oft signifikant hoch. In dieser Arbeit wird beschrieben, wie mit Hilfe einfacher Auswahlverfahren ein bifunktionales Protein designt werden kann. Anwendung findet diese Methode, indem eine DNA-Bindestelle in die Z-Domäne eingebracht wird. Die Struktur und die Fc-Antikörperbindestelle der Z-Domäne sollen dabei erhalten bleiben.

Die folgenden beiden Absätze führen in die informatischen wie auch biologischen Grundlagen ein, auf denen diese Arbeit aufbaut. Neben einem Absatz über genetische Algorithmen werden vor allem die Proteinbindestellen genauer beleuchtet, mit denen in diesem Projekt gearbeitet wird.

1.2 Genetische Algorithmen

Wenig Beachtung fanden 1858 Charles Darwin und Alfred Russel Wallace – wie viele Autoren zuvor – als sie ihre Arbeit [13] zur Evolutionstheorie in der Linnean Society of London vorstellten. Den Grundstein für das Verständnis und die Anerkennung der natürlichen Evolution legte Charles Darwin ein Jahr später, im November 1859, mit seinem Buch „On the Origin of Species“ [12]. Mit dieser Arbeit zeigte er, dass sich die natürlichen, komplexen Organismen durch wiederholtes Ausführen simpler Mechanismen aus einfacheren Organismen entwickelt haben. Dabei ist erstaunlich, wie effektiv sich verschiedene Organismen an ihre Lebenssituation anpassen konnten. Im Wesentlichen werden nur drei Mechanismen für diesen Evolutionsprozess genutzt:

die Rekombination von Erbgut durch Paarung zweier Individuen, die Mutation von Erbgut und die Selektion von Individuen auf Grund ihrer Überlebensfähigkeit.

„Die systematische Umsetzung der Prinzipien der Evolutionstheorie in computergesteuerte Optimierungssysteme erfolgte zunächst ab Mitte der sechziger Jahre vollkommen unabhängig voneinander an verschiedenen Stellen“ ([53], S. 358). Auf John H. Holland und David E. Goldberg ist die Entwicklung der Genetischen Algorithmen, auf Hans-Paul Schwefel und Ingo Rechenberg die der Evolutionsstrategien, auf John Koza die der Genetischen Programmierung und auf Lawrence J. Fogel die der Evolutionären Programmierung zurückzuführen. Diese Methoden fasst man zur Gruppe der Evolutionären Algorithmen zusammen. Obwohl die verschiedenen Optimierungsmethoden unterschiedlichen Ursprungs waren, ist ihnen auf Grund des gleichen, natürlichen Vorbildes der grundsätzliche Aufbau gemein. Ziel ist stets die Suche nach einer optimalen Lösung eines gegebenen Problems. Die Suche erfolgt iterativ durch wiederholtes Ausführen verschiedener Berechnungsschritte. Folgende Elemente lassen sich dabei für alle Evolutionären Algorithmen benennen:

- **Individuum** Ein Individuum repräsentiert eine mögliche Lösung für ein gegebenes Problem. Die innere Kodierung der Lösung ist problemszezifisch.
- **Population** Die Menge aller Individuen eines Evolutionären Algorithmus bildet eine Population.
- **Mutation** Die Mutation ist ein Operator, der auf ein Individuum angewendet wird und dieses verändert, in der Regel zufällig.
- **Rekombination** Die Rekombination ist ein Operator, der auf zwei oder mehrere Individuen angewendet wird. Er kombiniert die Lösungen dieser Individuen zu einem Satz neuer Lösungen. Somit werden neue Individuen, die so genannten Nachkommen, erstellt.
- **Fitness** Die Fitness eines Individuums beschreibt die Güte, mit der das Problem durch das Individuum gelöst wird.
- **Selektion** Die Basis für die Selektion eines Individuums bildet die Fitness. Die Selektion entscheidet an Hand der Fitness, welche Individuen für eine weitere Verarbeitung gewählt werden.
- **Generation** Als Generation wird eine Population beschrieben, die durch Selektion, Rekombination und Mutation berechnet wird. Durch die Anzahl der Generationen kann somit das Alter einer Population oder auch die Laufzeit des Algorithmus benannt werden.

Dabei unterscheiden sich die vier unterschiedlichen Evolutionären Algorithmen vor allem in der Art der Repräsentation einer Lösung im Individuum und in der konkreten Umsetzung der genetischen Operatoren: Mutation, Rekombination und Selektion.

Bekannt wurde der Genetische Algorithmus (GA) in den 70er Jahren, vor allem durch das Buch „Adaption in Natural and Artificial Systems“ [41] von John H. Holland. Aus der Gruppe der Evolutionären Algorithmen entsprechen die Strukturen eines GAs am meisten denen natürlicher Evolution und werden am häufigsten genutzt [53]. Die Individuen eines GAs werden in der Regel durch eine Zahlen- oder Zeichenkette repräsentiert. Die genetischen Operatoren (Rekombination und Mutation) agieren somit direkt auf dieser Zeichenkette.

1.2.1 Multiobjektive Optimierung mit Genetischen Algorithmen

Wie in vielen Fällen der Mathematik lässt sich ein frühes Optimierungsproblem auf die Griechen zurückführen. Der römische Dichter Vergil beschreibt in seinem Epos *Aeneis* die Gründung Karthagos: Nachdem die phönizische Prinzessin Dido vor ihrem Bruder geflohen war und 814 v. Chr. an der nordafrikanischen Küste landete, fragte sie den dortigen Häuptling nach so viel Land, wie sie mit einer Kuhhaut umspannen kann. Nachdem sie die Kuhhaut in dünne Streifen geschnitten hatte, bestand Didos Problem darin, die größte Fläche zu finden, die mit einer geschlossenen Kurve fester Länge umspannt werden kann, das so genannte isoperimetrische Problem [29].

Bei dem Problem von Dido handelt es sich um ein einobjektives Optimierungsproblem. Es gibt eine ideale Lösung für das gegebene Kriterium: Ein Kreis hat das beste Verhältnis von Umfang zur Fläche. Viele Probleme lassen sich jedoch nicht über ein Kriterium beschreiben. Werden mehr als ein Kriterium für ein Optimierungsproblem beschrieben, so handelt es sich um ein multiobjektives Optimierungsproblem.

Mehrdimensionale Probleme lassen sich auch mit klassischen Optimierungsverfahren lösen. Dazu muss zuerst das Problem in ein Eindimensionales kodiert werden. Eine solche Kodierung ist zum Beispiel die gewichtete Summe der einzelnen Kriterien. Dafür muss der Nutzer im Vorhinein eine Gewichtung der Kriterien festlegen. Man kann davon ausgehen, dass für dieses reduzierte Problem eine Lösung gefunden wird. Sie hängt jedoch von der Art der Kodierung ab. Soll eine weitere Lösung gefunden werden, muss die Gewichtung der Kodierungen geändert und die Optimierung wiederholt werden.

Als Beispiel soll hier der Entscheidungsprozess für den Kauf eines Autos dienen (vgl. Abb. 1.1). Die Kriterien sind der Preis und die Motorleistung des Autos. Einzeln betrachtet kann das Ideal dieser beiden Kriterien erreicht werden, so ist etwa ohne

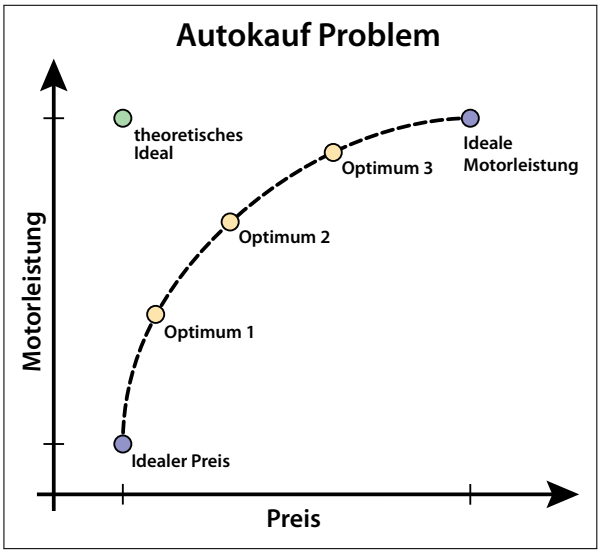


Abbildung 1.1: Mehrkriterielles Autokaufproblem. Blau: Ideale Lösungen der einzelnen Kriterien *Preis* und *Motorleistung*. Grün: Theoretische, ideale Lösung beider Kriterien. Gelb: Verschiedene optimale Lösungen nach einer mehrkriteriellen Optimierung auf der Pareto-Front (gestrichelte Linie). Aus [14], S. 2, abgeändert.

Rücksicht auf die Motorleistung das günstigste Auto schnell gefunden. Das gemeinsame, theoretische Ideal kann jedoch nicht erreicht werden [14]. Diese Struktur von Problemen verdeutlicht die Vorteile von Verfahren, bei denen schon der eigentliche Optimierungsprozess mehrdimensional abläuft.

Gegenüber numerischen Optimierungsverfahren haben die GAs bei dieser Art von Problemen den Vorteil, eine ganze Menge von Lösungen in einer Population vorzuhalten. Durch diese Menge von Individuen kann der Suchraum an mehreren Stellen gleichzeitig durchsucht werden. Während der Optimierung wird so nicht mehr nur zu einem Ziel hin optimiert. Ergebnis sind mehrere optimale Lösungen, die so genannte Pareto-Front (vgl. Abs. 1.2.2). Die für den Benutzer beste Lösung zu finden liegt nun bei ihm selbst. Im Autokaufbeispiel kann auf Grund von persönlichen Vorlieben oder Bedürfnissen des Käufers aus der Menge der optimalen Lösungen gewählt werden. Natürlich existieren weitere Lösungen in der Population von Individuen, die nicht optimal sind und somit nicht auf der Pareto-Front liegen, jedoch für verschiedene genetische Operatoren weiter vorgehalten werden.

1.2.2 Pareto-Front

Um die multiobjektive Optimierung mit Genetischen Algorithmen zu verbinden, muss zuerst der Begriff *Pareto-Front* genauer erläutert werden. Als Pareto-Front bezeichnet

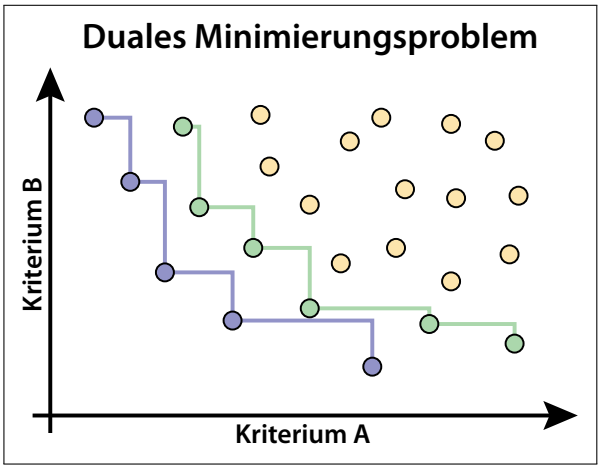


Abbildung 1.2: Visualisierung eines dualen Minimierungsproblems. Dargestellt ist eine Population von Individuen (Kreise) aufgetragen nach deren Kriterien sowie die Pareto-Fronten des Ranges eins (blau) und zwei (grün).

man die Elemente einer Menge, deren Kriterien mindestens gleich gut sind wie die jeweiligen Kriterien aller anderen Elemente und in mindestens einem Kriterium besser sind. Für den eindimensionalen Fall ist diese Relation einfach zu errechnen, da eine Ordnung im Zahlenraum existiert: Es gilt etwa die Relation $>$ oder $<$ der rationalen Zahlen. Für den mehrdimensionalen Fall ist es jedoch nötig, eine spezielle Ordnung zu definieren. Sind nun die Kriterien Elemente eines Vektors, so wird die Dominanz von Vektoren für Maximierungsprobleme definiert durch: Ein Vektor $\vec{a} = (a_1, \ldots, a_n)$ von Kriterien dominiert einen Vektor $\vec{b} = (b_1, \ldots, b_n)$ wenn gilt:

$$a \succ b := \exists i \in \{1, \ldots, n\}, \text{so dass} \, \forall \, k \neq i \, \text{gilt}: \; a_i > b_i \, \text{und} \, a_k \geqslant b_k \tag{1.1}$$

$a \succ b$ liest sich: „a dominiert b“. Für Minimierungsprobleme lässt sich die Dominanz durch Umdrehen der Relationen $>$ und $\geqslant$ im hinteren Teil der Formel 1.1 definieren, da hier kleinere Kriterien als besser gelten. Mit Hilfe der Dominanz von Vektoren lässt sich nun die Pareto-Front leicht definieren als Menge aller Elemente, die von keinem anderen Element der Menge dominiert werden. Abbildung 1.2 verdeutlicht eine solche Pareto-Front für ein zweidimensionales Minimierungsproblem.

Durch die Einführung verschiedener multiobjektiver evolutionärer Algorithmen, etwa der MOGA von Fonseca und Fleming [27], der NSGA von Srinivas und Deb [79] oder der NPGA von Horn et al. [42], wurde klar, dass für die genetischen Operatoren ein Operator von Nöten ist, der ein mehrdimensionales Problem auf ein Eindimensionales abbildet [15]. Dieser Operator kann mit Hilfe der Dominanz definiert werden: der Pareto-Rang. Nach der Berechnung der Pareto-Front wird allen Elementen auf

dieser Front der Pareto-Rang 1 zugewiesen. Danach werden diese Elemente aus der Menge entfernt. Nun wird die Pareto-Front der verbleibenen Elemente erneut berechnet und den Elementen der neuen Pareto-Front der Pareto-Rang 2 zugewiesen. Dieser Schritt wird wiederholt, bis kein Element mehr in der Menge vorhanden ist. Durch Zuweisung eines Ranges zu jedem Element können die Elemente im Anschluss sortiert werden und so eindimensionale genetische Operatoren, speziell die Selektion, weiter genutzt werden.

1.3 Z-Domäne von Protein A

In der Bioinformatik werden Themen der Informatik mit Themen der Biologie vereint. Da der informatische Teil dieser Arbeit mit einer Einleitung zu Genetischen Algorithmen bereits behandelt wurde, folgt nun für den biologischen Teil eine Übersicht über die in dieser Arbeit verwendeten Proteine.

Das Bakterium *Staphylococcus aureus* produziert ein etwa 42 kDa schweres Protein namens Protein A (SpA). Es ist ein Typ-1 Membranprotein, lokalisiert auf der Zellwand des Bakteriums, und besteht aus fünf homologen Domänen E, D, A, B und C gefolgt von einer zellwandbindenen X-Domäne (vom N-Terminus zum C-Terminus) [88]. Die fünf Domänen, alle etwa 58 Aminosäuren lang, zeigen eine starke Bindung zu Immunglobulinen IgG, IgA und IgM verschiedener Säugetiere, darunter auch Menschen. Die Bindung erfolgt an den Fc-Teil und nicht an die antigenbindene Region Fab der Immunglobuline [28, 54]. Sie stellen für *Staphylococcus aureus* einen Pathogenitätsfaktor dar: Durch die Bindung an Fc wird eine effektive Opsonierung verhindert und das Bakterium so vor phagozytierenden Zellen des Immunsystems geschützt.

Die guten Bindeeigenschaften von SpA an Fc sind interessant für die Forschung. So wurden weitere biologische Eigenschafften von SpA bekannt, wie etwa die Bindung an Fab [89], die Aktivierung der Synthese polyklonaler Antikörper [77] oder das Anregen der Interferon-Produktion [67]. Es folgten strukturbiologische Untersuchungen an SpA, vor allem an der B-Domäne. Deisenhofer et al. analysierten die Kristallstruktur der B-Domäne im Komplex mit Fc von IgG bei einer Auflösung von 2,8 Å [16]. Dabei konnte die Struktur der nicht im Kontakt zu Fc stehenden Helix nicht ausreichend genau aufgelöst werden. Später zeigte eine Kernspinresonanz-Spektroskopie (NMR-Spektroskopie) die B-Domäne in Lösung mit drei antiparallelen α-Helices [87] (vgl. Abb. 1.3).

Die Z-Domäne ist ein von der B-Domäne von SpA abgeleitetes Protein. Neben einer Mutation von Glyzin zu Alanin an Position 29 besitzt die Z-Domäne die gleiche Proteinsequenz wie die B-Domäne [85]. Es wurde jedoch die DNA-Sequenz von SpA

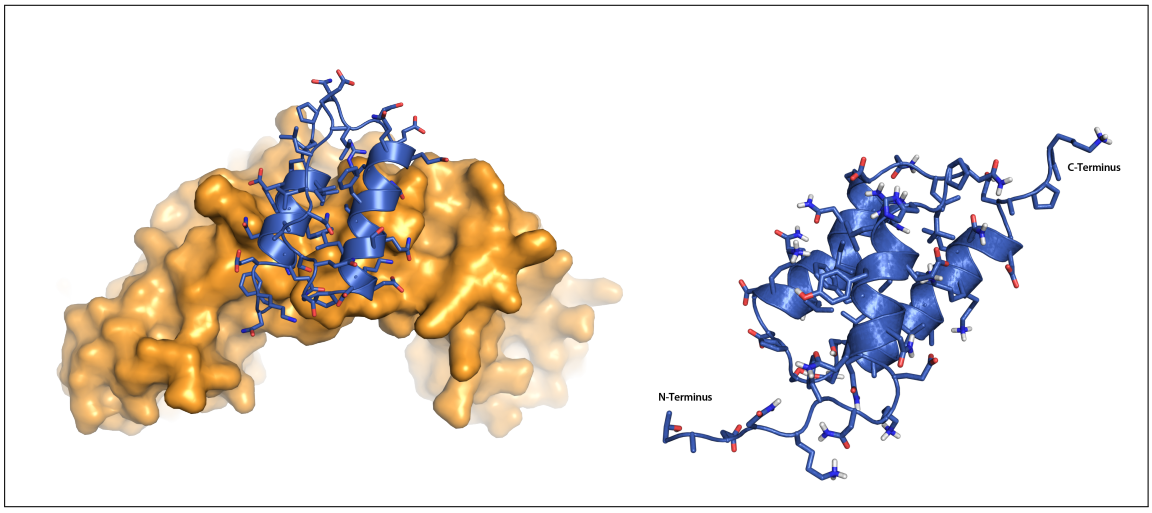


Abbildung 1.3: Kristall- und NMR-Struktur der B-Domäne (blau). Die Kristallstruktur (links) enthält zusätzlich den Fc-Teil von IgG (orange, Oberfläche). Die Strukturen wurden der Proteindatenbank [6] entnommen: `1FC2` links und `1BDD` rechts.

in einigen Teilen geändert, um eine einfachere Polymerisation des Proteins zu ermöglichen [63]. Viele Eigenschaften, wie die Bindeaffinität der Z-Domäne zu Fc, stimmen mit denen der B-Domäne überein [9, 54], was die Z-Domäne zu einem guten Modell zur Untersuchung der Bindung zwischen SpA und IgG macht.

In Abbildung 1.4 sind die an der Bindung an Fc beteiligten Aminosäuren in der B-Domäne abgebildet: Phe5, Gln9, Gln10, Asn11, Phe13, Tyr14, Leu17, Asn28, Ile31, Gln32 und Lys35 [16]. Die große Ähnlichkeit der Z-Domäne mit der B-Domäne lässt die Annahme zu, dass die an der Bindung beteiligten Aminosäuren der B-Domäne auch bei der Z-Domäne für die Bindung an Fc sorgen.

1.4 MyoD

Der Myogene Faktor 3 (MyoD oder auch Myf-3 beim Menschen) ist ein DNA-Transkriptionsfaktor im Zellkern von Tieren und gehört auf Grund seiner Struktur zu der Klasse der Helix-Loop-Helix-Proteine (bHLH, *basic-helix-loop-helix*). Zusammen mit Myf-5, Myf-6 und Myogenin sorgt es für die Differenzierung von Fibroblasten zu Myoblasten, womit der Aufbau der Skelettmuskulatur eingeleitet wird [84, 91].

Die Struktur der bHLH-Proteine gibt ihrer Klasse den Namen. Sie bestehen aus einer kurzen α-Helix, die über eine flexible Schleife (engl. *loop*) mit einer langen α-Helix verbunden ist. Typischerweise enthält die längere Helix die DNA-Bindestelle mit mehreren basischen Aminosäuren. MyoD bindet, wie viele andere Proteine dieser Klasse, an die DNA-Sequenz `CANNTG` [7]. Die Basen der DNA wurden hier mit ihren Anfangs-

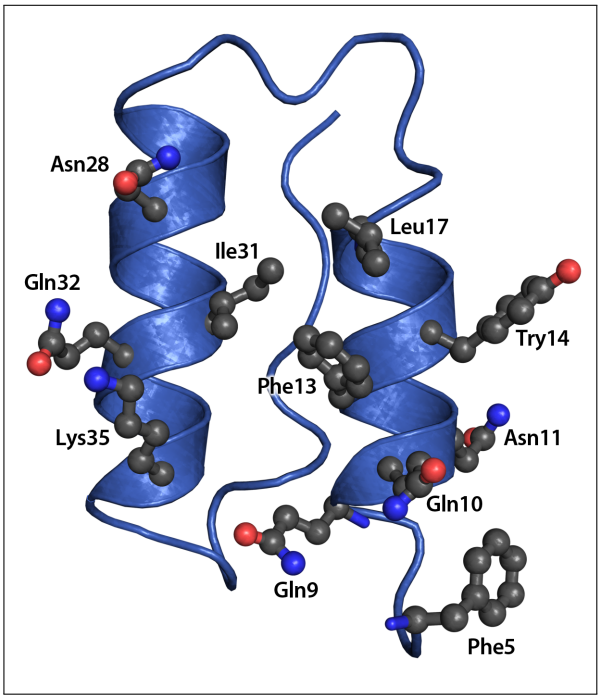


Abbildung 1.4: Kristallstruktur der B-Domäne im gebundenen Zustand (vgl. Abb. 1.3, links). Sicht auf die Fc-Bindestelle. Hervorgehoben sind die nach Deisenhofer an der Bindung beteiligten Aminosäuren [16] (aus [9], Seite 443, abgeändert).

buchstaben abgekürzt (**A**denin, **T**hymin, **G**lutamin und **C**ytosin). Das N steht für eine beliebige Base. Über die kürzere Helix kann MyoD andere bHLH-Proteine binden und formt so Homodimere mit einem weiteren MyoD-Molekül wie auch Heterodimere mit anderen bHLH-Proteinen [82]. Abbildung 1.5 zeigt eine Kristallstruktur eines MyoD-Homodimers gebunden an DNA.

Abbildung 1.6 ist eine schematische Darstellung der Interaktionen zwischen der DNA und der an der Bindung beteiligten Aminosäuren eines MyoD-Monomeres. Im Folgenden werden die Basen analog zur Abbildung 1.6 mit einem Index versehen und mit einem „*“ gekennzeichnet, wenn sich die Base auf dem Gegenstrang befindet.

Beide Monomere des Dimers bilden die gleichen chemischen Interaktionen zur DNA aus, jeweils auf dem entgegengesetzten Strang der DNA. Glu118 spielt bei der Bindung eine wichtige Rolle. Es bildet Wasserstoffbrückenbindungen (H-Brücken) zu den Basen von C5 und A4 und bindet über Wasser an die Basen von C8* und A4. Dabei wird die Seitenkette von Glu118 über eine Salzbrücke von Arg121 stabilisiert. Arg121 bindet das Phosphat von C5. Ebenfalls wichtige Interaktionspartner mit der DNA sind Arg111 und Thr115 der basischen MyoD-Region: Sie formen H-Brücken und hydrophobe Kontakte zur DNA sowie eine H-Brücke zueinander. Weitere Kontakte zu Phosphaten bilden Arg110, Arg117 und A119, die ebenfalls auf der basischen

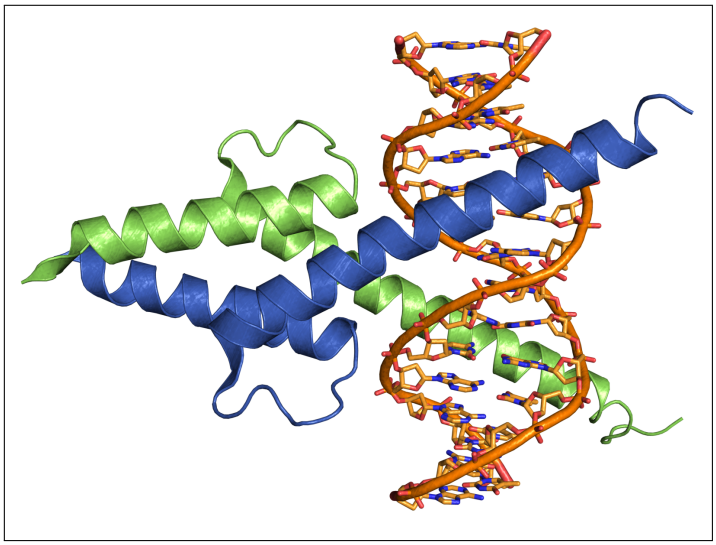


Abbildung 1.5: MyoD Homodimer (blau und grün) im Komplex mit DNA (orange). PDB-ID `1MDY`.

Binderegion liegen [56].

So sorgen Arg111, Thr115 und Glu118 durch Bindung an die Basen C5, A4 und ihren Gegenstücken auf dem Gegenstrang für die Spezifität der Bindung eines MyoD-Dimers an das Motiv `CANNTG`: Ein zweites MyoD-Molekül bindet gespiegelt um die Mitte der DNA (zwischen G7 und C8) an die Basen C5* und A4* respektive ihrer Gegenstücke. Zu den mittig liegenden Basen werden keine direkten Bindungen vermittelt.

1.5 Verwandte Arbeiten

Genetische Algorithmen wurden schon häufig zur Lösung bioinformatischer Probleme eingesetzt. Grund dafür ist vor allem ihre gute Performance bei komplexen Optimierungsproblemen und das hohe Maß an Adaptionsfähigkeit auf fast jedes Problem. Als besonders interessant zeigen sich in diesem Zusammenhang die Arbeiten von Gronwald et al. [34] und Scott et al. [74]. In beiden Arbeiten wird die Stabilität eines Proteins durch eine Optimierung mittels Genetischer Algorithmen erhöht, jedoch unterscheiden sich die eingesetzten Fitnessfunktionen fundamental.

Gronwald et al. [34] optimieren das 36 Aminosäuren lange Villin Kopfstück. Interessant bei diesem Ansatz ist die kleine Anzahl an Individuen und Generationen für eine GA-Optimierung. Über 15 Generationen mit je 8 Individuen konnte die stabile Wildtypsequenz aus der instabilen Mutante F18K wiederhergestellt werden. Weiterhin enthielten die Autoren als Ergebnis Sequenzen, die eine höhere Stabilität als der Wildtyp aufweisen. Trotz der geringen Anzahl an Optimierungsschritten kam der

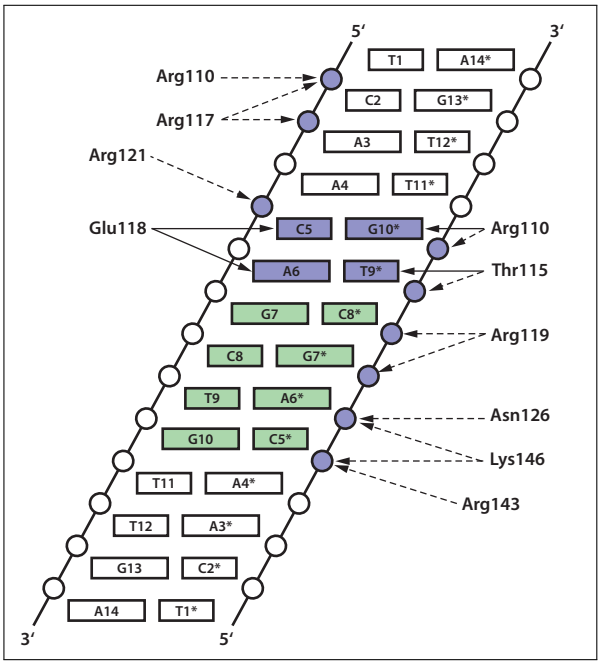


Abbildung 1.6: Bindeinteraktionen zwischen DNA und MyoD. Die Phospate der DNA werden als Kreise, die Basen als Rechtecke angezeigt. Die Basen des Bindemotivs von MyoD `CANNTG` sind blau und grün gekennzeichnet. Bindungen der MyoD-Aminosäuren zu den Phosphaten der DNA sind gestrichelt, zu den Basen durchgezogen dargestellt. An der Bindung beteiligte Elemente der DNA sind blau gekennzeichnet (aus [56], Seite 454, abgeändert).

verwendete Genetische Algorithmus auf eine Rechenzeit von etwa einem Jahr. Grund dafür ist die aufwändig zu berechnende Fitnessfunktion: Es wurden 10ns Molekulardynamiksimulationen eingesetzt, um die Stabilität einer jeden Struktur zu bestimmen.

Die Gruppe um Scott [74] nutzt für die Optimierung einen anderen Ansatz. Die Grundidee, ein Protein durch Optimierung der Sequenz mittels GA zu stabilisieren, ist jedoch ähnlich: Der hydrophobe Kern des Proteins Cytochrome B, bestehend aus 16 Aminosäuren, soll stabilisiert werden. Die Fitnessfunktion nutzt zur Berechnung der freien Energie des hydrophoben Kerns eine vereinfachte Version des van der Waals Terms. Das Rückgrad des gesamten Proteins bleibt fixiert, ersetzt werden lediglich die Seitenketten der 16 Aminosäuren des Kerns durch Seitenketten aus einer Bibliothek mit verschiedenen Orientierungen von Aminosäureresten. Durch die drastische verringerte Komplexität der Fitnessfunktion verkürzt sich die Rechenzeit des GAs bei 200 Individuen und über 100 Generationen auf wenige Minuten.

Obwohl Molekulardynamiksimulationen durch ihre hohe Genauigkeit schnell gute Ergebnisse liefern, lässt sich durch das Ersetzen der Bewertungsfunktion durch eine geeignete van der Waals Berechnung der Rechenaufwand stark reduzieren.

1.6 Forschungsmotivation und Aufbau der Arbeit

Die im vorherigen Kapitel vorgestellten Arbeiten verdeutlichen die Möglichkeiten, die durch Sequenzoptimierungen mittels Genetischer Algorithmen möglich sind. Diese Arbeit wird zeigen, dass es durch Reduzierung der Komplexitäten in der Fitnessfunktion möglich ist, bei angemessenem Rechenaufwand auch Modelle größerer Proteine im Ganzen zu optimieren. Folglich wird eine baukastenähnliche Konstruktion von Proteinen ermöglicht. Nach der Vorgabe einer Struktur und nur der wichtigsten Aminosäuren an bestimmten Positionen werden Sequenzen generiert, die in die gegebene Struktur falten und somit die erwünschte Funktion erfüllen. Die Findungsphase eines oder mehrerer erfolgsversprechender Kandidaten wird durch dieses Herangehen automatisiert.

Für die Biotechnologie, Chemie, Medizin und Pharmazie ersetzt eine solche Simulation und Optimierung eines Proteins nicht die Arbeit im Labor, die Funktionen und Strukturen von Molekülen zu testen. Durch den Einsatz immer schnellerer Computersysteme könnten jedoch mit dieser Methode Kandidaten für stabile Proteine mit den gewünschten Bindeeigenschaften gefunden werden. Langwierige und teure Laboruntersuchungen werden dadurch auf ein Minimum reduziert und das Finden von spezifischen Proteinen erleichtert.

In Kapitel 2 wird an Hand eines Fallbeispiels die Funktionalität der Methode erläutert. Hinzukommend werden alle Methoden erklärt, die für dieses Projekt zum Erzeugen und zur Analyse der Ergebnisse genutzt werden. Im Anschluss werden die ersten Ergebnisse in mehreren Schritten überprüft und die Ergebnisse diskutiert. Basierend auf den in Kapitel 2 gewonnen Erkenntnissen beschäftigt sich Kapitel 3 mit der Optimierung der Methode in verschiedenen Punkten. Somit soll die Leistung der Methode für das gestellte Problem maximiert werden. In Kapitel 4 folgt eine Zusammenfassung der gesamten Arbeit und ein Ausblick auf mögliche Anwendungen und weitere Verbesserungen.

2 Design eines DNA- und Fc-bindenen Fusionsproteins

«Science, at bottom, is really anti-intellectual. It always distrusts pure reason, and demands the production of objective fact.»

Henry Louis Mencken

2.1 Einleitung

Mit Hilfe aktueller Entwicklungsmethoden wird eine generische Softwareumgebung entwickelt. Definierte Schnittstellen und ein strukturiertes Softwarekonzept erleichtern die Implementierung von neuen Funktionen oder gänzlich unterschiedlichen Evolutionsprozessen. Die hohe Geschwindigkeit der aktuellen JAVA-Laufzeitumgebungen sowie die Verfügbarkeit einer großen Menge an Bibliotheken führen zu der Entscheidung, JAVA 1.6 [66] als Programmiersprache zu wählen. Die Verfügbarkeit verschiedener Entwicklungs- und Testprogramme für JAVA ermöglichen eine effektive Programmierung.

Einen Überblick über den gesamten Designprozess gibt die Abbildung 2.1. Nach der Konstruktion der Startsequenz (Abs. 2.2) wird ein Genetischer Algorithmus (GA)

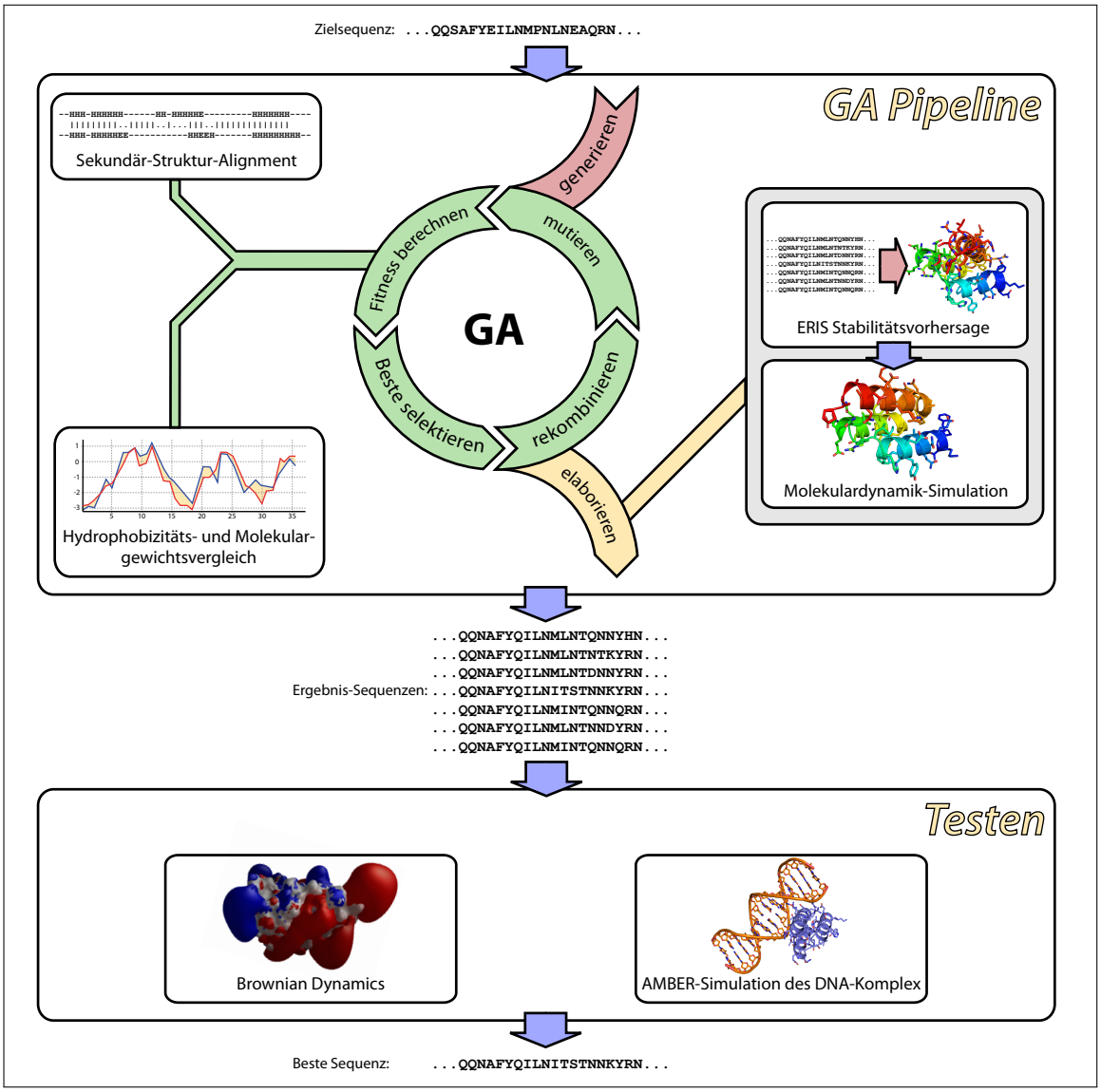


Abbildung 2.1: Übersicht über den gesamten Optimierungsprozess. Im ersten, automatisierten Teil (oben) wird die Optimierung mit Hilfe eines GAs durchgeführt. Anschließend wird eine Vorauswahl der Ergebnisse aus dem Optimierungsprozess durch ERIS und eine GROMACS Simulationen getroffen. Im zweiten Teil (unten) werden nach einer manuellen Auswahl die Sequenzen mittels Brownian Dynamics und AMBER Simulationen genauer geprüft.

initialisiert und gestartet (Abs. 2.3 und 2.4). Mit Hilfe verschiedener bioinformatischer Methoden wird das Ergebnis des GAs nachbearbeitet und verfeinert (Abs. 2.5). Im Anschluss wird die aussichtsreichste Sequenz basierend auf weiteren Tests aus dem Ergebnis der Optimierung gewählt (Abs. 2.6).

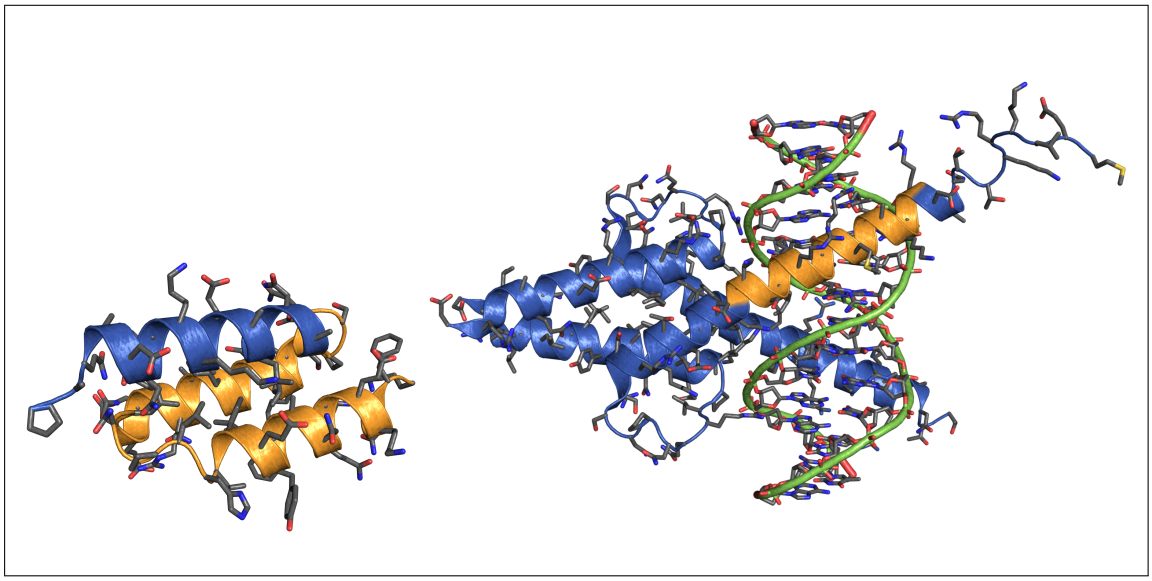


Abbildung 2.2: Strukturen der Z-Domäne (links) und des MyoD-Homodimers gebunden an DNA (rechts). Die DNA im rechten Komplex ist grün dargestellt. Orange markiert sind die Aminosäuren, die zum Erstellen der Startsequenz für den GA genutzt werden (vgl. Abb. 2.3).

2.2 Fusionierung der Z-Domäne mit einer DNA-bindenen Helix aus MyoD

Als Anwendungsbeispiel der Sequenzoptimierung soll eine DNA-bindene Helix in die Z-Domäne eingebracht werden. Die Strukturen für das folgende Vorgehen wurden der *Protein Data Bank* (PDB) [6] entnommen. Verwendet wird Kette B aus dem Eintrag `1LP1` [40] für die Z-Domäne und der Eintrag `1MDY` [56] mit einem Protein-DNA-Komplex von MyoD (vgl. Abb. 2.2). Die C-Terminale dritte Helix der Z-Domäne wird durch die DNA-bindene Helix von MyoD ersetzt. Es muss darauf geachtet werden, dass die für die Bindung an die DNA relevanten Aminosäuren nach diesem Prozess weiterhin nach außen zeigen und nicht in den Kern der Z-Domäne ragen. Die genaue Auswahl an Aminosäuren und deren Platzierung wird in Abbildung 2.3 gezeigt. Diese Sequenz ist sogleich die Startsequenz für den GA (vgl. Abs. 2.3).

Neben dem Einbringen einer neuen Funktion in die Z-Domäne soll eine der ursprünglichen Funktionen erhalten bleiben: die Fc-Bindestelle auf den Helices 1 und 2. Es ist offensichtlich, dass die essentiellen Aminosäuren der beiden neuen funktionalen Regionen für die Bindung an die DNA und Fc zwingend nötig sind und Mutationen dieser Aminosäuren die jeweilige Bindestelle zerstören können (vgl. Abs. 1.3 und 1.4). Aus diesem Grund werden während der Optimierung die in Abbildung 2.3 markierten Aminosäuren nicht verändert. Zusätzlich sind für Mutationen im Optimierungspro-

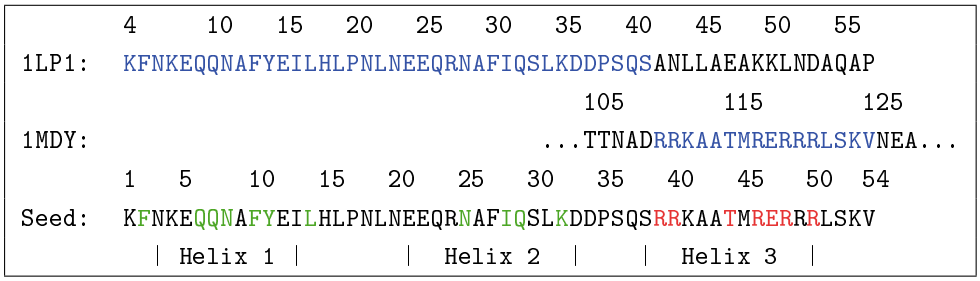


Abbildung 2.3: Konstruktion der Startsequenz für den GA. Blau markiert sind die Aminosäuren der Z-Domäne und MyoD, die in der Startsequenz übernommen werden. Die während der Optimierung konservierten Aminosäuren sind grün und rot markiert: Grün die Fc-bindenen Aminosäuren der Z-Domäne, rot die DNA bindenen von MyoD.

zess nur Aminosäuren zulässig, die in der Startsequenz vorkommen, um sterische Probleme zu vermeiden und den Optimierungsprozess zu vereinfachen.

Durch dieses Vorgehen soll weniger ein biologisch relevantes Protein erzeugt, als viel mehr die Leistung des Optimierungsverfahrens geprüft werden: Die transferierte Helix muss in MyoD anderen Ansprüchen gerecht werden als ein Teil der Z-Domäne zu sein. Neben der ungleichen Nettoladung der Z-Domäne (negativ geladen) und der DNA-bindenen Helix (positiv geladen), ist letztere umgeben von Zellplasma. An ihrer neuen Position muss vor allem ein hydrophober Bereich zur Innenseite der Z-Domäne gerichtet ausgebildet werden, um eine korrekte Faltung sicher zu stellen. Kann der GA mittels der verwendeten einfachen Fitnessfunktionen aus der gegebenen Aufgabe die Sequenz für ein stabiles Protein generieren, lassen sich mit hoher Wahrscheinlichkeit auch weniger komplexe Optimierungen erfolgreich durchführen.

2.3 Genetischer Algorithmus

Schon seit etwa 20 Jahren werden Genetische Algorithmen (GAs) immer wieder erfolgreich für das Proteindesign eingesetzt [17, 44]. Die Methode ist jedoch keinesfalls veraltet [92]. Gerade weil die Funktionsweise eines GAs stark an die natürliche Evolution angelegt ist, eignet er sich besonders zur Optimierung von Proteinsequenzen. Einen geeigneten Evolutionsdruck auf die Sequenzen auszuüben, ist hier jedoch nicht trivial. Dieser Druck wird durch die Fitnessfunktion ausgeübt, die für jede Sequenz in einer Generation berechnet wird.

Die Konzeption eines GAs gibt vor, erst mit einer großen Zahl von Individuen über viele Generationen hinweg effektiv zu simulieren. Das macht jedoch eine häufige Berechnung der Fitnessfunktion nötig. Sind diese Fitnessfunktionen zeitaufwändig in der Computerberechnung, summiert sich die Berechnung auf eine extrem lange Laufzeit

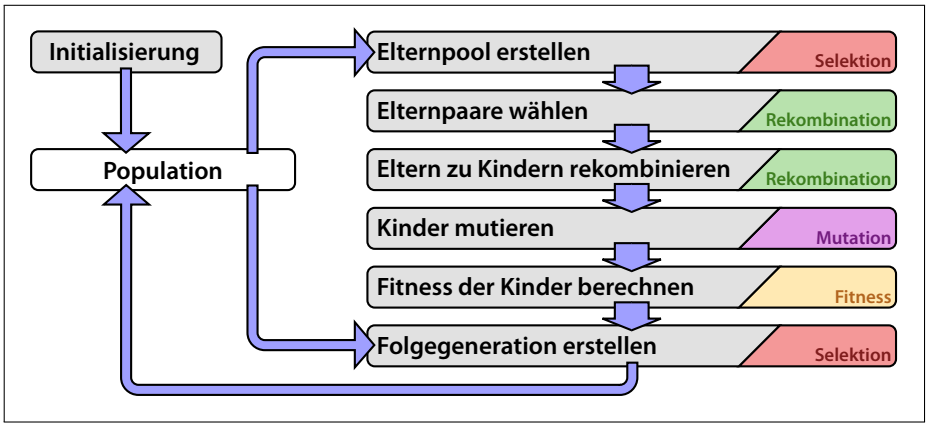


Abbildung 2.4: Detaildarstellung des internen Ablaufs eines GAs. Grau dargestellt sind verschiedene Operationen. Die Individuen werden in einer Population gespeichert. Eine Generation wird mit dem Durchlaufen der rechten Kette von Operationen abgeschlossen.

des Algorithmus. Eine 30 Sekunden dauernde Fitnessberechnung addiert sich bei 1000 Generationen mit 400 Individuen auf circa 140 Tage. Die Fitnessfunktion muss deshalb eine geringe Laufzeit bei der Berechnung haben und trotzdem einen geeigneten evolutionären Druck auf die Sequenzen ausüben, um das Simulationsziel zu erreichen. Die verwendeten Fitnessfunktionen werden in Absatz 2.4 genauer behandelt.

Den inneren Aufbau des GAs zeigt Abbildung 2.4. Nach der Initialisierung der ersten Elterngeneration von Individuen und der Berechnung der Fitness dieser Generation folgt der iterative Optimierungsprozess. Nach der Rekombination und anschließender Mutation der durch die Rekombination neu entstandenen Individuen, wird die Fitness dieser so genannten Kindergeneration berechnet. Mit einer Selektion werden geeignete Individuen der Eltern- und Kindergeneration zur Elterngeneration des nachfolgenden Iterationsschrittes vereinigt. Dieser Prozess wird wiederholt, bis die gewünschte Anzahl an Generationen erreicht wird.

Der Parametersatz des GAs aus diesem Kapitel beruht auf Erfahrungen aus einem vorangegangen, ähnlichen Projekt [1]. Es werden zwei Simulationen mit 600 Individuen über 1000 respektive 2000 Generationen gestartet. An den Ergebnissen soll beurteilt werden, ob eine Simulation über 1000 Generationen lang genug ist oder 2000 Generationen nötig sind, um ein gutes Simulationsergebnis zu erreichen. Die Mutationsrate beträgt dabei 1%. Die Wahl der genetischen Operatoren entspricht denen klassischer Ansätze. Besprochen wird die Optimierung dieser Parameter in Kapitel 3.

Das konkrete Ziel dieser Optimierung ist eine Sequenz mit gegebenen chemischen und physikalischen Eigenschaften zu finden, die in die Struktur der Z-Domäne faltet. Die Eigenschaften sind in diesem Fall folglich die der Z-Domäne. Verwendet werden

sekundärstrukturelle Merkmale und die Hydrophobizität beziehungsweise das Molekulargewicht. Da die Fitnessfunktionen vergleichend arbeiten, ist deren Referenz die Sequenz der Z-Domäne, im Weiteren Zielsequenz genannt. Durch die Optimierung soll das Ungleichgewicht, entstanden durch die in Helix 3 eingebrachten Aminosäuren der DNA-Bindestelle, in den genannten Eigenschaften ausgeglichen werden.

Initialisierung Mit der Initialisierung des GAs beginnt die Optimierung. Aus 600 zufällig erzeugten Individuen wird die erste Generation erstellt. Die nicht zu verändernden Aminosäuren der Startsequenz werden in die neuen Individuen übernommen. Die restlichen Aminosäuren werden gleichverteilt zufällig gewählt. Hierzu werden nur Aminosäuren genutzt, die auch in der Zielsequenz vorkommen. So werden unter anderem Aminosäuren mit großen, hydrophoben Seitenketten ausgeschlossen. Die Optimierung wird dahingehend vereinfacht, dass zum einen weniger Aminosäuren für den Algorithmus zur Verfügung stehen und die Komplexität sinkt. Zum anderen ähneln sich Hydrophobizitäts- und Molekulargewichtsvorhersage mehr denen der Z-Domäne. So ist Tryptophan nicht für die Optimierung verfügbar: Zum Ausgleich eines hydrophoben Kernes könnte es in der Kernregion platziert werden und durch die große Seitenkette später zu sterischen Problemen und folglich zu Instabilitäten im Kern führen.

Zusätzlich zur Erzeugung der ersten Generation werden die Fitnessfunktionen initialisiert. Hierzu werden die Fitnesswerte der Zielsequenz zu sich selbst errechnet und gespeichert. Somit entfällt eine Neuberechnung der Vergleichswerte der Zielsequenz bei jedem Aufruf der Fitnessfunktion für die einzelnen Individuen (vgl. Abs. 2.4).

Berechnung der Fitness Die Fitness eines jeden Individuums des GAs setzt sich aus drei Werten zusammen: der vorhergesagten Sekundärstruktur, der Hydrophobizität und dem Molekulargewicht. Details zu der Berechnung der einzelnen Fitnesswerte folgen in Absatz 2.4. Nachdem die Fitness bestimmt wurde, wird der Pareto-Rang wie in Absatz 1.2.2 berechnet. Dieser wird als zusätzlicher Fitnesswert gespeichert und für genetische Operatoren genutzt, die eine eindimensionale Fitness voraussetzen, etwa die Selektion.

Selektion Die Selektion geschieht im GA an zwei Stellen. Zu Beginn eines Optimierungslaufes wird eine Menge von Individuen aus der Generation ausgewählt, um per Rekombination eine neue Generation zu erzeugen. Diese Menge von Individuen wird im Weiteren *Mating Pool* genannt. Die Auswahl basiert auf einem gewichteten Auswahlverfahren. Die einzelnen Individuen werden an Hand ihres Pareto-Ranges ge-

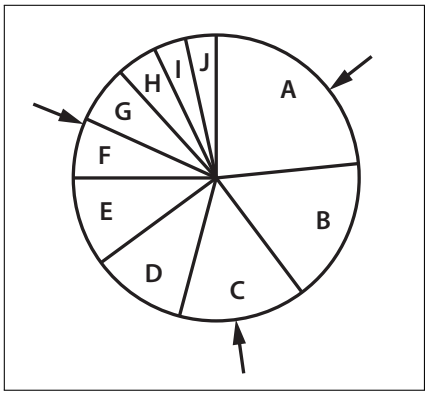


Abbildung 2.5: Roulette-Selektion von drei Individuen. Die Individuen A–J nehmen eine unterschiedlich große Fläche ein, abhängig von ihrer Pareto-Fitness. Selektiert werden nach dem Drehen die drei Individuen, auf die die Pfeile zeigen.

wählt. Individuen mit Pareto-Rang 1 haben eine hohe Wahrscheinlichkeit gewählt zu werden, solche mit einem hohen Rang eine eher geringe Wahrscheinlichkeit: Sei n_p die Anzahl der Pareto-Fronten $\mathcal{P} = \{\mathcal{P}_1, \ldots \mathcal{P}_n\}$ der Ränge 1 bis n, so ist die Pareto-Fitness der Individuen auf der Pareto-Front k: $f_{\text{pareto}}(\mathcal{P}_k) = 1 - (k/n_p)$. Die Wahrscheinlichkeit, ein Individuum auszuwählen, ist dann proportional zur Pareto-Fitness des Individuums. Diese Art der Selektion wird häufig Roulette-Selektion genannt. Die Anordnung aller Individuen im Kreis gewichtet nach ihrer Pareto-Fitness erinnert an ein Glücksrad. Für den Auswahlprozess wird dieses Rad gedreht. Um n Individuen zu erhalten, werden n Zeiger in gleichem Abstand um das Rad positioniert (vgl. Abb. 2.5).

Die zweite Stelle, an der selektiert wird, ist beim Zusammenfügen der Elterngeneration mit der neu erzeugten Nachkommengeneration am Ende eines Optimierungsschrittes. Da zu diesem Zeitpunkt zwei vollständige Populationen vorliegen, müssen diese auf eine Population reduziert werden. Dabei wird die Anzahl der Individuen wieder auf die Soll-Populationsgröße halbiert. Die Individuen werden an Hand ihres zuvor festgelegten Pareto-Ranges ausgewählt. Nur die besten Individuen werden in die neue Population aufgenommen.

Rekombination Der so genannte *One Point Crossover* wird als Operator für die Rekombination verwendet. Aus zwei zufällig aus dem *Mating Pool* gewählten Individuen werden zwei neue Individuen erzeugt. Dazu werden die Seqenzen ab einer zufällig gewählten Stelle miteinander getauscht (vgl. Abb. 2.6).

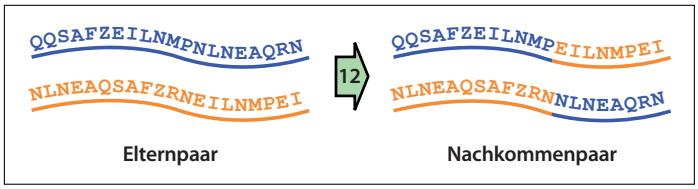


Abbildung 2.6: Der *One Point Crossover* Operator angewandt auf zwei Elternsequenzen. Durch Tausch an der Position 12 werden zwei Nachkommen erzeugt.

Mutation Nach der Rekombination wird jedes Individuum mutiert. Für jede Aminosäure eines Individuums wird mit einer Wahrscheinlichkeit von 0,01 eine neue Aminosäure zufällig gewählt. Die nicht zu verändernden Aminosäuren sind von der Mutation ausgeschlossen. Es werden keine zusätzlichen Aminosäuren eingefügt oder gelöscht. Die Sequenzlänge des Individuums bleibt erhalten.

2.4 Fitnessfunktionen

Die Fitnessfunktionen für den GA sind das kritischte Element für den Erfolg einer Optimierung. Weil eine hohe Anzahl von Individuen und Generationen oft unerlässlich für ein gutes Ergebnis sind, muss im Bereich des Protein-Designs auf eine kurze Laufzeit der Funktionen geachtet werden. Hinzukommend war ein wichtiger Anspruch an die Fitnessfunktionen die Möglichkeit des verteilten Rechnens auf mehreren Computersystemen. Die im Folgenden vorgestellen Fitnessfunktionen erfüllen diese Eigenschaften.

Die drei verwendeten Fitnessfunktionen lassen sich in zwei Kategorien aufteilen: Zwei Funktionen arbeiten auf der Primärstruktur-Ebene, also der Aminosäurensequenz. Die dritte Funktion arbeitet auf der Sekundärstruktur-Ebene. Für die Ebene der Primärstruktur wurde eine Hydrophobizitäts- und eine Molekulargewichtsvorhersage gewählt. Die Sekundärstrukturvorhersage wird mit Hilfe von GAME-SSP [3] durchgeführt. In vorherigen Studien konnten wir zeigen, dass die Kombination einer Hydrophobizitäts- und einer Sekundärstrukturvorhersage wichtige Eigenschaften von Proteinen beschreiben kann [37]. Durch Hinzunahme des Molekulargewichtes sollen gegebenenfalls sterische Probleme vermieden werden.

2.4.1 Sekundärstrukturvergleich

Die Sekundärstrukturvorhersagen wurden mit Hilfe von GAME-SSP durchgeführt, einer Implementierung von PSIPRED [45] basierend auf dem GAME-Framework [3]. Zur Schonung von Ressourcen wurden die in Python geschriebenen Scripte in Ja-

va übersetzt und in das bestehende Framework integriert. GAME-SSP nutzt, wie auch PSIPRED, positionsspezifische Bewertungsmatrizen vom Basic Local Alignment Search Tool (BLAST). In der ursprünglichen Implementierung wurden diese Matrizen aus einer Anfrage an einen BLAST Server gewonnen. Dieser Aufruf wurde aus der Software entfernt und durch einen Aufruf eines auf dem lokalen Rechner installierten BLAST Programmes ersetzt. Erst diese Änderung erlaubt das verteilte Rechnen auf mehreren Computern. Ein einzelner Webserver könnte die große Anzahl an Anfragen nicht ausreichend schnell verarbeiten.

Um die Vorhersagegeschwindigkeit weiter zu steigern, wurde die von BLAST zu benutzende Datenbank geändert: UniRef90 [83] wurde durch SWISS-PROT [4] ersetzt. Die Dauer eines BLAST-Suchlaufes sinkt etwa um den Faktor 100. Die Vorhersagegenauigkeit von GAME-SSP verringert sich laut der Autoren jedoch um 1-4%. Zu Gunsten der Performance ist diese Verringerung hinnehmbar.

Zum Erstellen eines konkreten Fitnesswertes aus der Sekundärstrukturvorhersage wird die Ähnlichkeit der Vorhersage eines jeden Individuums zu der Vorhersage der Zielsequenz mittels eines Sequenzalignments bestimmt. Hierzu wurde die Open-Source-Software JAligner [62] verwendet. Bei JAligner handelt es sich um eine JAVA-Implementierung des Smith-Waterman-Algorithmus [78] mit Optimierung durch den Gotoh-Algorithmus [32] zum paarweisen, lokalen Sequenzvergleich. Um die Laufzeit des Algorithmus zu verringern, wurden Dateizugriffe auf die Substitutionsmatrizen minimiert und die Zielsequenz, welche immer gleich bleibt, nur beim erstmaligen Starten des Algorithmus eingelesen.

Für das Vergleichen von Sekundärstruktursequenzen wurde eine neue Substitutionsmatrix entwickelt, da ein Sekundärstrukturalignment andere Elemente enthält als ein Sequenzalignment: Strukturmerkmale anstatt Aminosäuren. Übereinstimmende Strukturen bekommen dabei einen hohen Wert von 8. Liegt keine Übereinstimmung vor, wird -4 als Wert gesetzt (vgl. Tab. 2.1). In einem früheren Projekt hat sich gezeigt, dass Sekundärstrukturalignments mit dieser Substitutionsmatrix die phylogenetische Hierarchie von Sequenzen, die mit herkömmlichen Sequenzalignments bewertet wurden, beibehält [1]. Der von JAligner ermittelte Alignmentscore wird im Folgenden als Sekundärstruktur-Fitnesswert genutzt.

2.4.2 Hydrophobizitäts- und Molekulargewichtsvergleich

Für die Vorhersage der Hydrophobizität nach dem Kyte-Doolittle [49] Verfahren wird jeder Aminosäure ihr Hydrophobizitätswert zugeordnet. Eine Auflistung der verwendeten Werte befindet sich in Anhang A. Dieser Wert liegt zwischen -4,6 (hydrophile

	H	E	-
H	8	-4	-4
E	-4	8	-4
-	-4	-4	8

Tabelle 2.1: Substitutionsmatrix für JAligner zum Erstellen von Sekundärstrukturalignments. H: α-Helix, E: β-Faltblatt, -: Random Coil Struktur

Aminosäure) und 4,6 (hydrophobe Aminosäure). Die Berechnung erfolgt per *sliding-window* Verfahren: Es wird ein Fenster fester Breite über die Aminosäurensequenz geschoben. An jeder Stelle wird der Mittelwert der im Fenster liegenden Aminosäuren gebildet. Die Fensterbreite bestimmt dabei, wie fein das Verfahren auflöst. Für globuläre Proteine wird eine Fensterbreite von 5 bis 9 empfolen [49], um deren hydrophobe Eigenschaften zu beschreiben. Aus diesem Grund wurde in dieser Arbeit die Standardeinstellung des Algorithmus mit einer Fensterbreite von 7 beibehalten.

Zum Vergleich zur Zielsequenz sind noch weitere Schritte nötig. Wieder soll ein Wert die Ähnlichkeit beschreiben. Hierzu wird die Liste der Hydrophobizitätswerte als Graph interpretiert und mittels einer kubischen Spline Interpolation interpoliert. Diese Interpolationsart ist kaum rechenintensiv und bildet den Verlauf des Graphen naturgetreuer ab als etwa die lineare Interpolation [80]. Der resultierende Graph kann nun integriert und die Differenz zur Zielsequenz bestimmt werden. Sind $f(x)$ und $g(x)$ Graphen der Hydrophobizitätswerte, erhält man den Fitnesswert durch:

$$\int |f(x) - g(x)| \mathrm{d}x \tag{2.1}$$

Für die Integration wurde das numerische Verfahren der Rombergintegration [69] genutzt. Implementiert wurde die Interpolation und die Integration über eine externe Java Library. Hierzu wurde Apache Commons Math [86] eingebunden. Um Fehler durch unterschiedliche Längen der Sequenzen zu vermeiden, wird die kürzere Sequenz mit Nullen auf die Länge der langen Sequenz aufgefüllt.

Für die Berechnung des Molekulargewicht-Fitnesswertes wird das gleiche Verfahren verwendet. Jedoch werden die Hydrophobizitätswerte duch die Molekulargewichte der Aminosäuren nach Fasman [25] ersetzt. Die verwendeten Werte sind Anhang A zu entnehmen.

2.5 Elaboration

Wie schon in der Einleitung zu Genetischen Algorithmen erwähnt, erhält man als Ergebnis einer multikriteriellen Optimierung die Pareto-Front, eine Menge von jeweils optimalen Lösungen des Problems (vgl. Abs. 1.2.1). Diese Menge von gegebenenfalls über 100 Sequenzen soll automatisch untersucht und auf eine kleine Menge reduziert werden, um eine effektive manuelle Untersuchung zu ermöglichen. In einem ersten Schritt wird dazu mit Hilfe von ERIS [95] die Sequenz auf die Struktur der Z-Domäne modelliert und die Stabilität des resultierenden Modells bewertet. Danach werden die am besten bewerteten Modelle 20 ns in einer Molekulardynamiksimulation mit GROMACS [39] simuliert. Die Simulation mehrerer Modelle zur Steigerung der Vorhersagegenauigkeit ist wegen der hohen Rechenzeit nicht praktikabel.

2.5.1 Stabilitätsvorhersage mit ERIS

ERIS, benannt nach der griechischen Gottheit, ist ein Programm zur Vorhersage der Proteinstabilität. Es wurde von Yin et al. [94, 95] entwickelt und basiert auf der Molekularmodelling Suite Medusa [19]. So lässt sich im optimalen Fall eine Struktur für eine Aminosäurensequenz konstruieren, deren eigentliche Struktur noch nicht experimentell aufgeschlüsselt worden ist.

Das Modellieren einer Sequenz auf einer gegebene Struktur ist keinesfalls trivial. Bei der Modellierung kann es auf Grund von Differenzen in den Aminosäuren der Sequenz zu den Aminosäuren der Struktur zu sterischen, also räumlichen, Problemen kommen, da die Seitenketten der verschiedenen Aminosäuren unterschiedlich groß sind. Somit müssen diese Seitenketten in der Struktur neu gepackt werden, um eine physikalisch realistische Struktur zu erreichen. Außerdem müssen die flexiblen Schleifen in einer Proteinstruktur genauer betrachtet werden. α-Helices und β-Faltblätter bilden stabile Strukturen in einem Protein aus, die flexiblen Schleifen sind in ihrem Bewegungsspielraum gegebenenfalls sehr frei. Für diese Schleifen muss also ebenfalls eine möglichst optimale Position gefunden werden.

Neben der Modellierung einer Sequenz auf eine Struktur berechnet ERIS zusätzlich die Stabilität der neu erstellten Struktur in Relation zur Ausgangsstruktur. Dazu wird mit einem Kraftfeld, welches Parameter für alle Atome in der Struktur enthält, die chemische und physikalische Plausibilität dieser Struktur bewertet. Hohe Werte über Null geben dabei den Hinweis, dass die Aminosäurensequenz in der aktuellen Struktur instabil ist und sich gegebenenfalls entfalten kann. Werte unter Null sind ein Indiz für eine gefestigte Struktur.

ERIS wird in dieser Arbeit gegenüber anderen Programmen zur Stabilitätsvorhersage wie etwa FoldX [35] bevorzugt. Es liefert bei Vorhersagen mit ein bis zwei Mutationen vergleichbare Ergebnisse, wurde jedoch speziell entwickelt, um Mutationen von Aminosäuren mit kleinen zu größeren Seitenketten und Modelle mit hohem Anteil an Mutationen besonders gut vorauszusagen. Vor allem sterische Probleme bei diesen Mutationen werden durch eine Modellierung mit flexiblem Aminosäuren-Rückrad und nachfolgendem Neupacken der Seitenketten vermieden.

Diese Eigenschaften von ERIS decken sich gut mit den zu erwartenden Ergebnissen der Optimierung mit dem GA. Abgesehen von den Aminosäuren, die wegen Erhalt der Bindefähigkeit an Fc nicht mutiert werden dürfen, ist eine große Zahl von Mutationen zur Z-Domäne zu erwarten. Vor allem auf der dritten Helix mit der neuen MyoD Bindestelle werden Mutationen eingeführt, um den hydrophoben Kern des Proteins zu stabilisieren.

Auf Grund des Geschwindigkeitsvorteils wird ERIS lokal auf einem Rechner gestartet. Aufgerufen wird ERIS mit der Kristallstruktur der Z-Domäne (PDB Code: `1LP1`) und einer Liste der Mutationen, die nötig sind, um die Originalsequenz auf die Sequenz aus der GA-Optimierung zu mutieren. Zusätzlich bekommt ERIS als Parameter übergeben, das Aminosäurenrückrat flexibel zu modellieren und die Seitenketten der Aminosäuren neu zu packen. Als Ergebnis liefert ERIS die Stabilitätsvorhersage als einen Wert $\Delta\Delta G$ und Atommodelle der mutierten Sequenzen, die für spätere Simulationen genutzt werden. Eine zusätzliche Modellierung wird somit überflüssig. So können alle Sequenzen der letzten Pareto-Front des GAs nach Stabilität sortiert werden.

2.5.2 Molekulardynamiksimulation mit GROMACS

Nachdem die Sequenzen mit ERIS bewertet und dazugehörige Atommodelle konstruiert wurden, werden Molekulardynamiksimulationen (MD) von den zehn am besten und den zehn am schlechtesten bewerteten Modellen angefertigt. Außerdem wird zum späteren Vergleich der Strukturen die Startsequenz des GAs simuliert, die keine Optimierung durch den GA erfahren hat. Die *GROningen MAchine for Chemical Simulations* (GROMACS) ist ein ursprünglich an der Universität Groningen in den Niederlanden entwickeltes Softwarepaket für die Simulation molekulardynamischer Prozesse. Die mittlerweile unter der *GNU General Public License* stehende Software wird an vielen Orten weiterentwickelt. Die Modelle werden molekulardynamisch mit GROMACS 4.0.7 [39] simuliert.

Eine MD-Simulation berechnet die Bewegung von Atomen oder Molekülen in ei-

nem gegebenen Zeitabschnitt. Dazu werden die Newtonschen Bewegungsgleichungen aller Atome über die Zeit integriert. Die dabei auf die Atome oder Moleküle wirkenden Kräfte können unterschiedlich fein aufgelöst werden. Es ist offensichtlich, dass eine sehr genaue Berechnung aller auf ein Atom einwirkender Kräfte einen großen Rechenaufwand mit sich bringen. Solche Simulationen sind nur für einzelne Proteine in Lösung und einer kurzen Simulationszeit, hier 20 ns in vertretbarer Zeit möglich.

In den folgenden Simulationen werden die wirkenden Kräfte durch ein Kraftfeld beschrieben. Bei *GROMOS96 43a1* [75] handelt es sich nicht um ein Kraftfeld, welches Parameter für jedes einzelne Atom eines Systems beinhaltet, sondern um ein *united-atom* Kraftfeld [33]. Dabei werden verschiedene Gruppen von Atomen, etwa die Methylgruppe, als Ganzes parametrisiert. Das verringert die Komplexität und damit die Laufzeit der Berechnung. Diese Kraftfelder basieren auf molekularmechanischen Eigenschaften und sind geeignet, um physikalische Effekte wie Atombewegungen zu erkennen. Chemische Effekte, etwa das Auseinanderbrechen oder Neubilden von kovalenten Bindungen, sind mit diesen Kraftfeldern nicht zu reproduzieren.

Das in einem Kraftfeld beschriebene Potential E setzt sich im Fall von *GROMOS96 43a1* aus Energien für kovalente und nicht kovalente Wechselwirkungen sowie einen Therm für spezielle Wechselwirkungen zusammen:

$$E = E_{\text{kovalent}} + E_{\text{nichtkovalent}} + E_{\text{speziell}} \tag{2.2}$$

Dabei schlüsseln sich die physikalischen Energien weiter auf:

$$\begin{aligned} E_{\text{kovalent}} &= E_{\text{bindelänge}} + E_{\text{winkel}} + E_{\text{harmonisch}} + E_{\text{trigonometrisch}} \\ E_{\text{nichtkovalent}} &= E_{\text{lennard-jones}} + E_{\text{coulomb}} \end{aligned} \tag{2.3}$$

Die nicht physikalischen, speziellen Energien umfassen vor allem Terme zur Drosselung verschiedener Eigenschaften. Im Energieterm für die kovalenten Energien befinden sich Therme für die Bindelänge, Bindewinkel, und Dihedralwinkel. Die Energien der Dihedralwinkel werden je nach Art des Winkels über ein harmonisches Potential oder über eine trigonometrische Funktion beschrieben. Die nicht kovalenten Terme enthalten einen Teil zur Lösung des Lennard-Jones-Potential, womit vor allem Vander-Waals-Kräfte beschrieben werden, und einen Teil zur Coulombwechselwirkung, welche elektrostatische Energien beschreibt [75]. Abbildung 2.7 illustriert einen Auszug der Kräfte, für die Energien berechnet werden. Die Bindungen zwischen Atomen haben immer einen energetisch optimalen Zustand. Weicht eine Länge oder ein Winkel

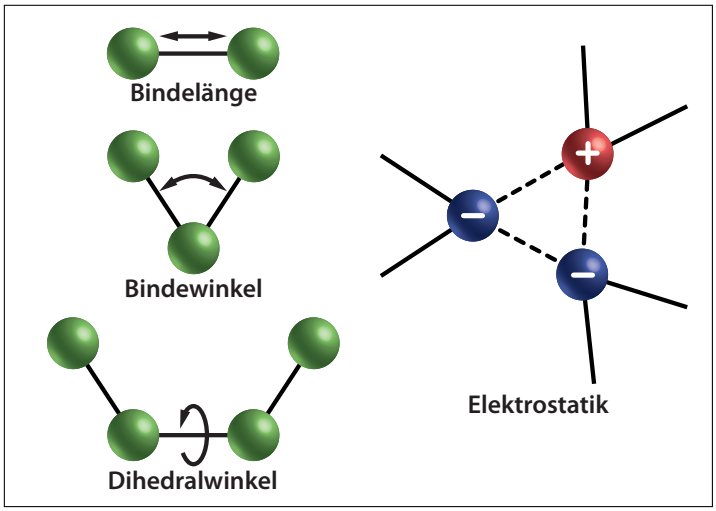


Abbildung 2.7: Schematische Darstellung verschiedener Kräfte in Molekülen (Kugeln), für die mit einem Kraftfeld Energien berechnet werden. Links: Kräfte in kovalenten Bindungen (durchgezogene Linie) für Bindelänge, Bindewinkel und Dihedralwinkel. Rechts: Kräfte einer nicht kovalenten Bindung (gestrichelte Linie), hier der Elektrostatik. Blau und Rot verdeutlichen negative respektive positive Ladungen.

von diesem ab, streben die Atome danach, den Optimalzustand wieder einzunehmen. Ist das nicht möglich, weil sie etwa im Raum durch andere Atome blockiert werden oder eine andere Kraft entgegenwirkt, wird das Molekül instabil. Die Energie des Systems nimmt zu.

Als Vorbereitung für die Simulation wird das Protein in einer kubischen Box platziert und diese mit Wasser gefüllt. Vor der 20 ns langen Produktionsphase, also der eigentlichen Simulation des Verhaltens des Moleküls, werden drei kurze Minimierungen durchgeführt. Diese dienen dazu, das hinzugefügte Wasser und das Protein, was aus einer Kristallstruktur stammt, im neuen Medium zu relaxieren.

Für die Minimierung wird als Iterator ein Verfahren der konjugierten Gradienten (cg) verwendet. Minimiert wird jeweils über 500 Schritte mit einer initialen Schrittweite von 0,01 nm. Die *Particle-Mesh Ewald* (PME) Methode wird mit einem Cutoff von 0,9 für die Elektostatik genutzt. Der Van-der-Waals Cutoff liegt bei 1,4 nm. Für die Minimierungen wird keine Temperaturkopplung oder Aufwärmphase definiert. Für alle verbleibenden Parameter werden die Standardwerte genutzt. Nacheinander wird für die Minimierung im ersten Schritt das Protein gefroren, danach nur noch dessen Rückrat. In der letzten Minimierung wird das komplette System minimiert.

Für die Produktionsphase wird in einem kanonischen Ensemble (NVT-Ensemble) über 20 ns simuliert. Als Integrator wird der *leap-frog* Algorithmus (md) mit einem Zeitschritt von 2 fs und somit 10.000.000 Schritten gewählt. Der Cutoff für die Elek-

trostatik und der Van-der-Waals Cutoff sind wie in der Minimierung gesetzt. Für die Temperaturkopplung wird Nose-Hoover mit einer Referenztemperatur von 300 K verwendet. Wasserstoffbrückenbindungen werden mit Hilfe des *linear constraint solver* (LINCS) behandelt.

Es ist zu erwarten, dass durch einen großen Anteil an Mutationen zum Zielprotein, die C-Terminale Helix mit der neuen DNA-Bindestelle der wichtigste Indikator für eine gelungene Optimierung ist. Zum einen wird hier angezeigt, ob der hydrophobe Kern der ursprünglichen Z-Domäne durch die Optimierung wieder hergestellt werden konnte. Ist dies nicht der Fall, könnte sich die Helix von den anderen beiden Helices lösen und sich das Protein teilweise entfalten. Die Helix könnte auch auf Grund zu starker Wechselwirkungen zu anderen Teilen des Proteins zwischen die beiden anderen Helices driften und so die Fc-Bindestelle zerstören. Ein leichtes Verdrehen der dritten Helix ist hinnehmbar, die helicale Struktur und die damit verbundene DNA-Bindefähigkeit der Helix muss jedoch erhalten bleiben.

Um dieses wichtige, strukturelle Merkmal zu untersuchen, wird nach der Simulation die Wurzel der mittleren quadratischen Abweichung (RMSD) der $C\alpha$-Atome der Aminosäuren auf der dritten Helix (Aminosäuren 39-53) über die Simulationsdauer hinweg berechnet. Als Referenz dient die initiale Struktur der Simulation. Ausgerichtet werden die einzelnen Schnappschüsse der Simulation über die $C\alpha$ der ersten beiden Helices (Aminosäuren 6-17 und 22-33). Der RMSD gibt dabei die geometrische Ähnlichkeit zweier Strukturen an und berechnet sich wie folgt: Seien s und u zwei Strukturen mit jeweils n Atomen. Jedes Atom besitzt dabei drei Koordinaten im Raum: x, y und z, so gilt:

$$\begin{aligned} \mathrm{RMSD}(s,u) &= \sqrt{\frac{1}{n}\sum_{i=1}^{n} \|s_i - u_i\|^2} \\ &= \sqrt{\frac{1}{n}\sum_{i=1}^{n} (s_{ix} - u_{ix})^2 + (s_{iy} - u_{iy})^2 + (s_{iz} - u_{iz})^2} \end{aligned} \tag{2.4}$$

Der RMSD wir in einer Längenangabe angegeben, meist in Nanometern oder Ångström (Å). Angewandt auf die Ergebnisse einer Simulation, ergibt der RMSD eine geometrische Abweichung für gegebene Atome über die Zeit der Simulation, in diesem Fall 20 ns. Um Aussagen über die Stabilität einzelner Regionen treffen zu können, wird außerdem die Wurzel der mittleren quadratischen Fluktuation (RMSF) berechnet. Anders als bei dem RMSD wird hier nicht über alle n Atome, sondern über die

Anzahl der Zeitschritte t der Simulation, summiert. Der RMSF beziehts sich dabei stehts auf ein einzelnes Atom s:

$$\begin{aligned} \mathrm{RMSF}(s) &= \sqrt{\frac{1}{t}\sum_{i=1}^{t} \|s_i - \tilde{s}\|^2} \\ &= \sqrt{\frac{1}{t}\sum_{i=1}^{t} (s_{ix} - \tilde{s}_x)^2 + (s_{iy} - \tilde{s}_y)^2 + (s_{iz} - \tilde{s}_z)^2} \end{aligned} \tag{2.5}$$

Zur Berechnung wird dabei ein Referenzpunkt $\tilde{s}$ durch Minimierung des RMSD aller Punkte s_i optimal superpositioniert. Als Ergebnis erhält man daraus die Fluktuation für jedes Atom. An Hand dieser Ergebnisse sollen die vielversprechendsten Sequenzen ausgewählt und mit weiteren Untersuchungen auf ihre Funktionalität getestet werden.

2.6 Manuelles Testen der Ergebnisse

Die bisherigen Prozesse und Methoden können voll automatisch ausgeführt werden und liefern am Ende eine kleine Anzahl von maximal zehn Sequenzen, aus denen mit Hilfe der Ergebnisse der GROMACS-Simulation eine Sequenz ausgewählt werden soll. Neben dem Erhalt der Fc-Bindestelle soll vor allem die Fähigkeit zur DNA-Bindung nach dem gesamten Optimierungs- und Auswahlprozess wie gewünscht ausgeprägt sein.

Die Fc-Bindestelle auf Helix 1 und 2 des neuen Fusionsproteins wird vor allem durch eine visuelle Inspektion im Vergleich zu der Struktur der Fc-Bindestelle mit der originalen Z-Domäne durchgeführt. Die neu eingebrachte DNA-Bindestelle lässt ein solch einfaches Vorgehen nicht mehr zu. Vorausgesetzt, die dritte Helix ist korrekt im Fusionsprotein ausgerichtet, so dass die konservierten Aminosäuren für die DNA-Bindung in das Medium ragen, müssen vor allem die DNA-Bindeeigenschaften überprüft werden. Langreichende Bindeeigenschaften werden über eine Brownian Dynamics Simulation mittels BrownDye [43] berechnet. Für die kurzreichenden Bindeeigenschaften wird eine MD-Simulation mit dem Programm AMBER [65] angefertigt.

2.6.1 Brownian Dynamics-Simulation

Als Brownsche Bewegung bezeichnet man die Bewegung eines Teilchens, etwa Atome oder Moleküle, in Flüssigkeit oder Gas. Grund für die Bewegungen sind die Atome

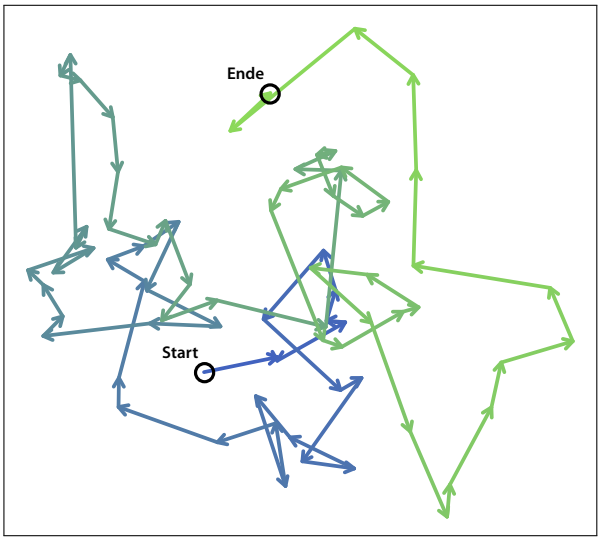


Abbildung 2.8: Darstellung einer brownschen Bewegung eines Moleküls im zweidimensionalen Raum. Wird das Molekül durch ein Atom des umgebenen Mediums getroffen, legt es eine gewisse Strecke in eine Richtung zurück (gekennzeichnet durch Pfeile). So wandert das Molekül auf einem zufälligen Pfad von seinem Startpunkt aus an einen zufälligen Endpunkt (von blau nach grün).

des umgebenen Mediums, die von allen Seiten gegen das Teilchen stoßen. Aus der Ursache der Bewegung ist leicht ersichtlich, dass das Ausmaß der Bewegung von der Temperatur sowie der Größe und Form des Teilchens abhängig ist. Wirkt keine weitere Kraft auf das Teilchen ein, so folgt es einer rein zufälligen Bewegung in alle drei Raumrichtungen (vgl. Abb. 2.8). Die Diffusion von Teilchen beruht auf dieser Bewegung.

In dem konkreten Fall dieser Arbeit soll untersucht werden, ob das optimierte Protein an ein Stück DNA bindet. Das setzt voraus, dass es durch gerichtete Diffusion zu der DNA gelangt. Ein rein zufälliges Zusammentreffen würde nicht zu einer ausreichend häufigen Bindung führen. Durch eine Simulation der Brownschen Bewegung des Proteins in Anwesenheit von DNA soll die Assoziationsrate berechnet werden, also die Wahrscheinlichkeit, mit der das neue Protein mit der DNA während der Simulation in Kontakt kommt. Ab einem gewissen Abstand des Proteins zu der DNA wird die für die Diffusion relevante Kraft durch das Coulombsche Gesetz beschrieben [64]. Sie wird durch Ladungsverteilungen auf den Oberflächen der beiden Moleküle erzeugt.

Mit der Software *BrownDye* [43] wird die Brownsche Bewegung wiederholt simuliert. Benutzt wird die Northrup-Allison-McCammon Methode [64]. Abbildung 2.9 verdeutlicht den Aufbau der Methode. Die DNA wird in der Mitte des Simulationsraumes platziert und fixiert. Der Raum um die DNA wird in zwei Bereiche unterteilt.

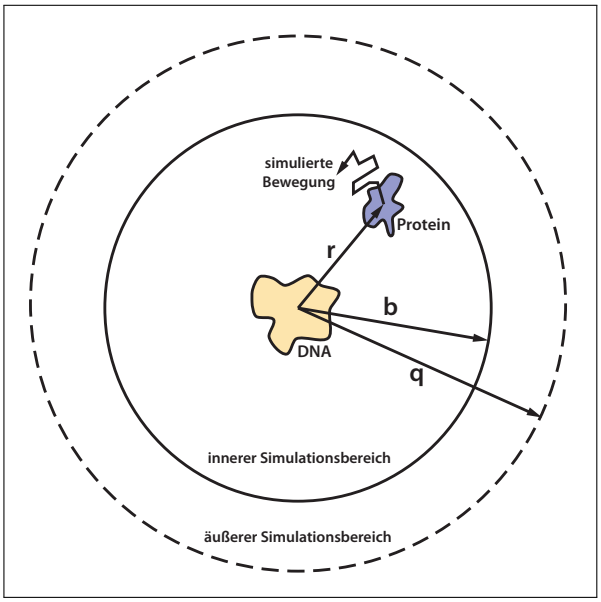


Abbildung 2.9: Zweidimensionale Darstellung der Northrup-Allison-McCammon-Methode [64] für Brownian Dynamics-Simulationen. Der Simulationraum wird aufgeteilt in einen inneren Bereich mit Radius b und einen äußeren Bereich mit Radius q. Wird der Abstand des Proteins zur DNA r größer als q, so wird die Simulation abgebrochen (aus [64], Seite 1518, abgeändert).

Auf der Grenze des inneren zum äußeren Bereich wird das Protein zu Beginn der Simulation zufällig platziert. Der Abstand des Proteins r ist somit gleich des Radius des inneren Bereichs b. Nun wird die Brownsche Bewegung mit den auf das Protein wirkenden Kräften berechnet.

Verlässt das Protein im Laufe der Simulation den äußeren Bereich, wird also sein Abstand größer als q, so wird die Simulation beendet. Das Protein hat in diesem Fall die DNA nicht gebunden. Zwei Atome der DNA und des Proteins werden als reaktives Paar bezeichnet, wenn sie eine polare Interaktion bilden und der Abstand der beiden unter 5,5 Å liegt. Liegen drei dieser reaktiven Paare nicht weiter als 5,5 Å auseinander, wird für diese Simulation eine erfolgreiche Assoziation angenommen. Ein Experiment setzt sich jeweils aus 25.000 Simulationen unter einer Salzkonzentration von 0,3 mol/l zusammen. Aus den Ergebnissen der Simulationen werden dann die Assoziationsraten für das Protein bestimmt. Die Radien b und q brauchen für *BrownDye* nicht angegeben werden. b wird automatisch berechnet auf Grund verschiedener Voraussetzungen, die b für eine Simulation erfüllen muss. Eine neue Version des Northrup-Allison-McCammon Algorithmus benötigt außerdem die Angabe von q nicht mehr.

BrownDye berechnet aus den 25.000 Simulationen eines Experimentes die absolu-

te Assoziationsrate (k_{on}) des Proteins an die DNA. Um die Simulationswerte besser bewerten zu können, wurden neben den Simulationen der Strukturen aus der Optimierung noch Negativ- und Positivtests angefertigt. Als Positivtest wird die aus MyoD entnommene DNA-bindene Helix zusammen mit der DNA simuliert. Als Negativtest wurde die Z-Domäne hergenommen, die keine DNA-Bindestelle besitzt. Außerdem wird als weiterer Anhaltspunkt die Struktur der Startsequenz des GAs nach der MD-Simulation mittels BrownDye simuliert. Auf Basis dieser Werte werden aus den absoluten Assoziationsraten die relativen Assoziationsraten (k_{on}^{rel}) berechnet.

Als Strukturen für die Simulation wird für die DNA die Struktur aus dem MyoD-Komplex gewählt (PDB-ID: `1MDY`). Ebenfalls aus diesem Komplex wird die Struktur der Positivkontrolle entnommen, für die die DNA-bindene Helix von MyoD herhält. Als Struktur für die Negativkontrolle wird die Struktur der Z-Domäne verwendet (PDB-ID: `1LP1`). Die Struktur der Startsequenz wird durch Modellieren der Startsequenz auf die Struktur der Z-Domäne erzeugt. Die Strukturen der restlichen Simulationskandidaten werden aus dem letzten Schritt der vorangegangenen MD Simulation gewählt.

2.6.2 AMBER-Simulation

Der erste Schritt, die Bindung des optimierten Proteins zur DNA zu untersuchen, ist eine Simulation des Proteins im Komplex mit DNA. Wie GROMACS (vgl. Abs. 2.5.2) ist AMBER [65] ein Programm zum Anfertigen von MD-Simulationen.

Die Simulationen werden mit *pmemd* aus AMBER 10 [8] angefertigt, als Kraftfeld wird *Amber99SB* genutzt. Der Protein-DNA-Komplex wird in einer oktaedrischen Box mit mindestens 10 Å Abstand zwischen Komplex und Rand der Box platziert. Die Box wird mit TIP3P Wasser gefüllt, einem 3-Punkt Wassermodell. Das System wird mittels des LEAP Modules mit Protonen in Gleichgewicht gebracht.

Durch den Einsatz des SHAKE-Algorithmus konnte ein Zeitschritt von 2 fs genutzt werden. Der SHAKE-Algorithmus hält die Längen von kovalenten Bindungen durch Eliminierung hoher Frequenzen konstant. Nach dem Entfernen hoher Frequenzen kann der Zeitschritt auf 2 fs erhöht werden, der jeweils von der höchsten Frequenz abhängt. Die Elektrostatik und die Van-der-Waals-Kräfte werde mit Hilfe der *Particle Mesh Ewald* (PME-Methode) berechnet. Der Cutoff-Wert liegt bei 9 Å. In der ersten Minimierung werden die Wassermoleküle und die Wasserstoffe in 100 Schritten mit dem Steilsten-Abstieg-Verfahren und danach in 100 Schritten mit dem Konjugierte-Gradienten-Verfahren minimiert. Anschließend wird das gesamte System mit den selben Methoden minimiert, jedoch mit jeweils 1000 Schritten. Vor der Produktions-

phase wird das System über 10 ps von 0 K auf 300 K in einem kanonischen Ensemble (NVT-Ensemble) aufgeheizt. Es folgt die Produktionsphase mit einer Länge von 10 ns in einem isothermisch-isobarischen Ensembles (NPT-Ensemble).

Der DNA-Protein-Komplex für diese Simulation wird aus der Kristallstruktur von MyoD (PDB-ID: `1MDY`) konstruiert. Der erste der beiden DNA-Doppelstränge wird für die Simulation als Rezeptor gewählt. Anschließend wird die Struktur des optimierten Proteins als Ligant an die DNA positioniert. Dies geschieht durch Ausrichten der während der Optimierung für die DNA-Bindung konservierten Aminosäuren auf der dritten Helix des Proteins zu den korrespondierenden Aminosäuren auf der DNA-bindenen Helix von MyoD. Verwendet wird die Struktur des Proteins aus dem letzten Simulationsschritt der GROMACS-Simulation. Der Protein-DNA-Komplex wird mit PyMOL [73] erstellt und anschließend in zwei einzelne Strukturen aufgeteilt, den Liganden und den Rezeptor. Für die Ausrichtung des Proteins an die DNA werden die $C\alpha$-Atome der Aminosäuren 39, 40, 44, 46-48 und 50 an die der entsprechenden Aminosäuren der MyoD Struktur ausgerichtet (vgl. Abs. 1.4 und Abb. 2.3).

Die Seitenketten der DNA-Bindestelle des optimierten Proteins werden vor der Simulation nicht umpositioniert und nehmen im Komplex so gegebenenfalls eine unkorrekte, nicht optimale Position ein. Dies verschlechtert unter Umständen die Ergebnisse der Simulation, jedoch sind Proteine mit nicht ausreichend konservierter DNA-Bindeleistung dadurch einfacher zu identifizieren.

2.7 Ergebnisse

2.7.1 Sequenzoptimierung mittels Genetischer Algorithmen

Wie in Absatz 2.3 wurden mit dem vorgestellten GA zwei Optimierungen mit 1000 respektive 2000 Generationen durchgeführt (im Folgenden als Gen1000 beziehungsweise Gen2000 bezeichnet). Als Ergebnis dieser Läufe wurden zwei Pareto-Fronten errechnet. Die Pareto-Front der Gen1000-Optimierung besteht aus 67 Individuen. Die der anderen beinhaltet 86 Individuen auf der Pareto-Front der letzten Generation.

Abbildung 2.10 zeigt das Konvergenzverhalten der Optimierungen an Hand der Pareto-Fronten jeder Generation. Beide Läufe zeigen nach 20% der Simulationszeit bezogen auf die drei berechneten Fitnesswerte keine relevante Änderung mehr. Kleine Änderungen sind jedoch bis etwa 60-70% der Simulationszeit zu beobachten. Es lässt sich also festhalten, dass beide Optimierungsläufe während der Simulationszeit konvergiert sind.

Dieses Bild ist typisch für die genutzten genetischen Operatoren (vgl. Abs. 2.3).

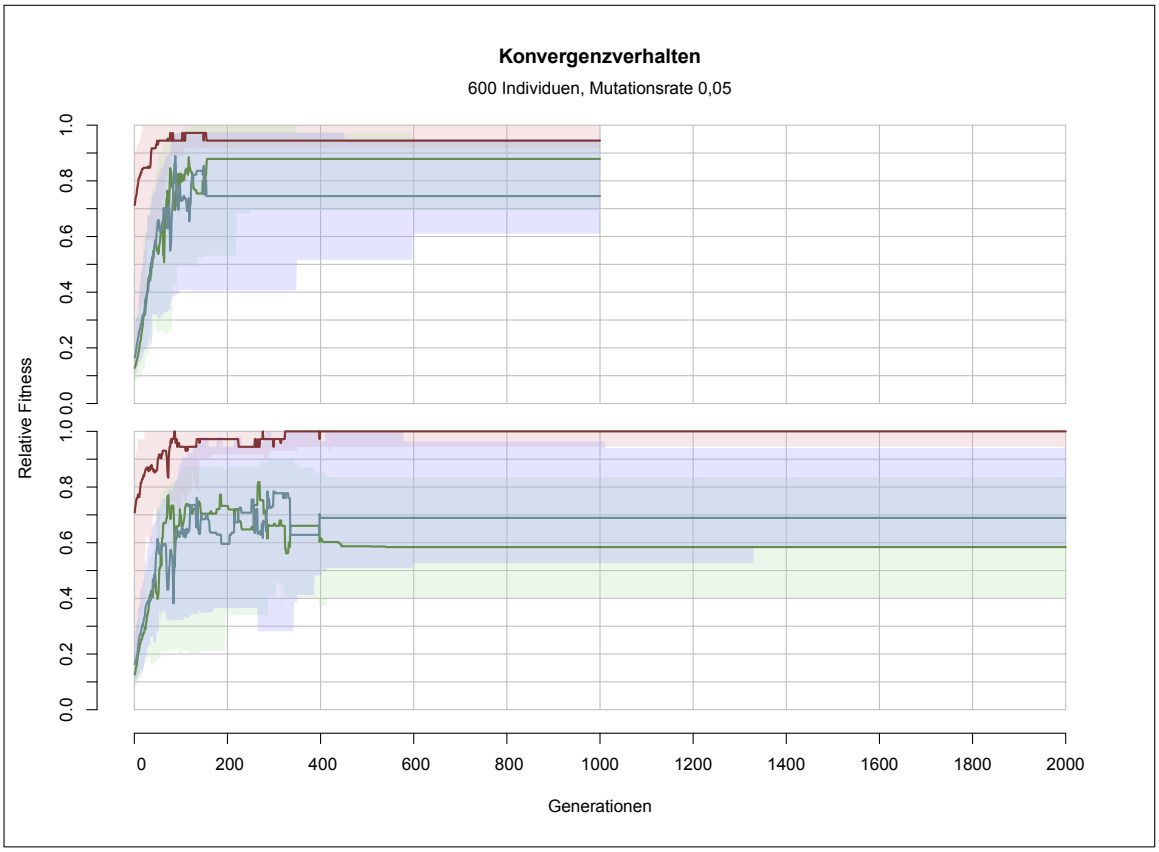


Abbildung 2.10: Konvergenzverhalten beider GA-Optimierungen an Hand der drei Fitnesswerte der Individuen auf der Pareto-Front einer jeden Generation. Die dargestellten Werte sind auf das jeweilige Maximum beider Simulationen normiert und auf den Bereich zwischen 0 und 1 skaliert. Oben ist die Optimierung über 1000, unten die über 2000 Generationen gezeigt. In Rot dargestellt ist der Sekundärstruktur-Wert, in Grün der Hydrophobizitäts- und in Blau der Molekulargewichts-Wert. Die farbige Linie gibt den mittleren Fitnesswert an, der farbige hellere Bereich markiert das Minimum und Maximum der Fitnesswerte.

Der in dieser Konfiguration gewählte elitäre Ansatz beim Zusammenfügen der Elternpopulation mit der Nachkommengeneration ermöglicht eine rasche Konvergenz in ein Minimum der Fehleroberfläche des gestellten Problems. Änderungen an der Pareto-Front sind nach einer ersten Konvergenz eher selten, da vorhandene gute Individuen in der Population nicht verloren gehen.

Bei dem Vergleich der beiden Läufe zueinander fällt ebenfalls auf, dass sich die Fitnesswerte, verglichen mit dem korrespondierenden Wert des anderen Laufes, schnell auf ein unterschiedliches Niveau eingependelt haben und dieses im Laufe der restlichen Optimierung (jeweils nach etwa 20% der angesetzten Optimierungsschritte) nicht mehr verlassen. Dies legt nahe, dass beide Optimierungen zwar konvergiert sind, jedoch nicht das selbe Optimum gefunden haben. Mindestens eine der beiden Opti-

mierungen hat also das globale Optimum des Problems nicht erreicht. Auf Grund des elitären Ansatzes ist es sogar wahrscheinlich, dass das globale Optimum des Problems nicht gefunden wurde. Diese Annahme stützt sich auf der extrem rauhen Fehleroberfläche des gegebenen Problems: Schon eine Mutation in der Sequenz kann entweder stabilisierend wirken oder die Faltung des Proteins komplett zerstören.

Abbildung 2.11 verstärkt die Annahme, dass zwei unterschiedliche Optima gefunden wurden. Gezeigt wird ein Sequenzalignment der Sequenzen aus den Pareto-Fronten der letzten Generationen beider Optimierungsläufe in der *WebLogo*-Darstellung [11]. Neben den, während der Optimierung konservierten Aminosäuren, stimmen vom N-Terminus über die erste Helix bis etwa zur Mitte der ersten Schleife (Aminosäuren 1-15) beide Sequenzalignments überein. Die Bereiche der ersten Schleife bis zur Mitte der zweiten Helix (Aminosäuren 16-24) und der Schleife zwischen der zweiten und dritten Helix (Aminosäuren 33-37) unterscheiden sich stark und weisen innerhalb ihrer Population eine hohe Diversität auf. Gemein ist beiden Optimierungen, dass zwischen konservierten Aminosäuren der dritten Helix vor allem Alanin und Valin eingesetzt wurden. Somit wird der hydrophilen Wirkung des polaren, positiv geladenen Arginins entgegengewirkt und der hydrophobe Kern des Proteins durch die eingesetzten Aminosäuren auf der dritten Helix stabilisiert.

Das Problem, die dritte Helix in das neue Milieu einzupassen, wurde während beider Optimierungsläufe auf dieselbe Weise gelöst. Stellen mit hoher Diversität sind vor allem in den Schleifen zwischen den Helices zu finden, Regionen, die auch in der Natur sehr flexibel gegenüber Mutationen sind.

2.7.2 Bewertung der Sequenzen mit ERIS

Basierend auf der mit ERIS erstellten Rangfolge wurden die zehn am besten und die zehn am schlechtesten bewerteten Sequenzen beider Optimierungen ausgewählt. Anhang B listet diese Sequenzen sortiert nach ihrer errechneten Stabilität ($\Delta\Delta G$) auf. Die Sequenzen der Gen1000 Optimierung zeigen insgesamt geringere $\Delta\Delta G$-Werte als die Sequenzen der Gen2000 Optimierung: im Durchschnitt 9,73 zu 16,44. Die $+/-$ Werte hingegen sind bei der längeren Optimierung geringer: 2,64 im Gegensatz zu 3,64. Die Verteilung innerhalb der Population ist dabei etwa gleich. *ERIS* bewertet somit die Strukturen der kürzeren Optimierung insgesamt als stabiler, gibt jedoch für die Strukturen der längeren Optimierung eine höhere Verlässlichkeit der Ergebnisse an.

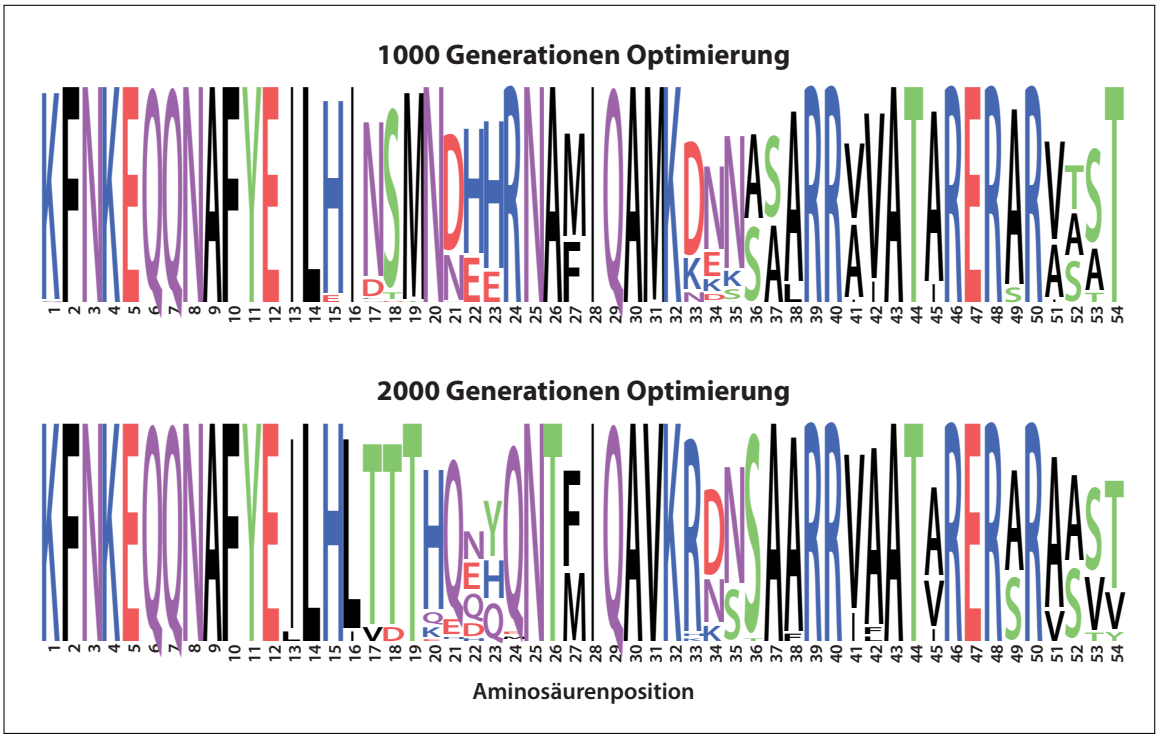


Abbildung 2.11: Logo-Darstellung der Aminosäuren aller Sequenzen der jeweils letzten Pareto-Front aus den beiden GA-Optimierungen. Die Buchstaben kodieren die Aminosäuren, die Höhe der Buchstaben gibt die Häufigkeit des Vorkommens der zugehörigen Aminosäure in dem Satz von Sequenzen der Pareto-Front an. Die Aminosäuren sind nach ihren physikalischen Eigenschaften farbkodiert: grün: polar, violett: neutral, blau: basisch, rot: sauer, schwarz: hydrophob. Erstellt mit WebLogo [11].

2.7.3 GROMACS-Simulation der Top und Flop 10

Nach den 40 Simulationen über je 20 ns mit GROMACS wurde der RMSD und RMSF wie in Absatz 2.5.2 beschrieben berechnet. Abbildung 2.12 zeigt eine Gegenüberstellung der RMSD-Werte der beiden Optimierungsläufe, berechnet wie in Absatz 2.3 beschrieben an Hand der $C\alpha$-Atome der dritten Helix. Um die Übersicht in den Grafiken zu wahren, wurden die insgesamt 4001 RMSD-Werte einer jeden Simulation mittels einer Spline-Interpolation mit 40 Stützstellen geglättet. Der RMSD der Gen2000-Optimierung zeigt keinen Unterschied zwischen den am besten und am schlechtesten von *ERIS* bewerteten Sequenzen. Die Werte der Gen1000-Optimierung sind im Gegensatz dazu besser separiert. Insgesamt weisen diese jedoch größere Werte auf. Während der Simulation verhalten sich die Strukturen der Gen1000-Simulation also insgesamt instabiler in der dritten Helix im Vergleich zu den Strukturen der Gen2000-Simulation. Das Ergebnis der Stabilitätsvorhersage mit *ERIS* wird jedoch nur von den Gen1000-Simulationen reproduziert.

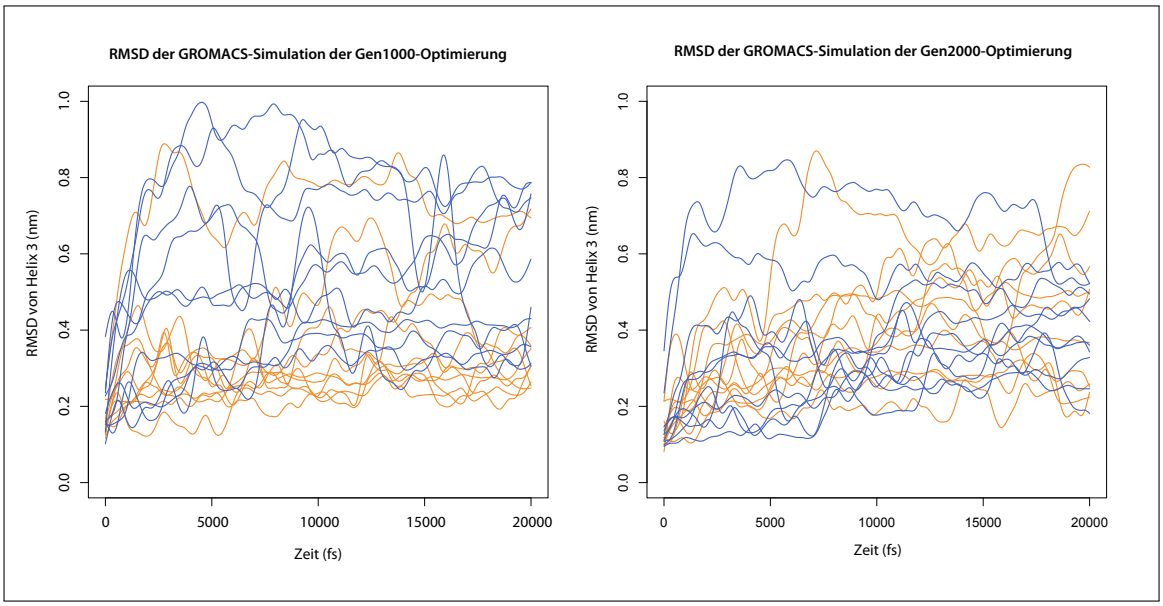


Abbildung 2.12: RMSD der GROMACS-Simulationen berechnet mit dem Programm `g_rms_d` aus dem GROMACS-Paket [39] wie in Absatz 2.5.2 beschrieben. In orange die Strukturen, die als beste von ERIS bewertet wurden, in blau die am schlechtesten bewerteten. Auf der horizontalen Achse ist die Simulationszeit und auf der vertikalen ist der RMSD in nm aufgetragen.

In Abbildung 2.13 dargestellt ist der RMSF der Strukturen aus der GROMACS-Simulation. Dieser bestätigt die Annahme einer geringeren Stabilität der Strukturen der Gen1000-Optimierung. Vor allem die Bereiche von Atom 90–120, etwa der Bereich der zweiten Schleife zwischen der zweiten und dritten Helix des Proteins, weist im Gegensatz zu den Gen2000-Strukturen höhere Werte auf. Somit bewegen sich diese Bereiche während der Simulation stärker und verlassen gegebenenfalls ihre Startkonformation. Ziel war es jedoch, genau diese zu erhalten. Ein hoher RMSF spricht also für ein Nichterreichen des Optimierungszieles.

Aus der Erfahrung mit MD-Simulationen ist bekannt, dass der mathematische RMSD und der RMSF nur Hinweise zur Stabilität von Proteinen während einer MD-Simulation bieten kann. Aus diesem Grund wurden zusätzlich alle 40 Simulationen manuell untersucht. Dabei wurde vor allem das Verhalten des Proteins über den Simulationszeitraum hinweg, der Drift zwischen der ersten und zweiten Helix mit der Fc-Bindestelle und eine eventuelle Repositionierung der dritten Helix, betrachtet.

Wie der RMSD und RMSF der Gen1000-Simulationen vermuten ließ, zeigten die Strukturen während der Simulation eine höhere Instabilität. Für die am besten und am schlechtesten bewerteten Strukturen wurden gleichermaßen Repositionierungen der dritten sowie der ersten und zweiten Helix beobachtet. Weiterhin waren die Struk-

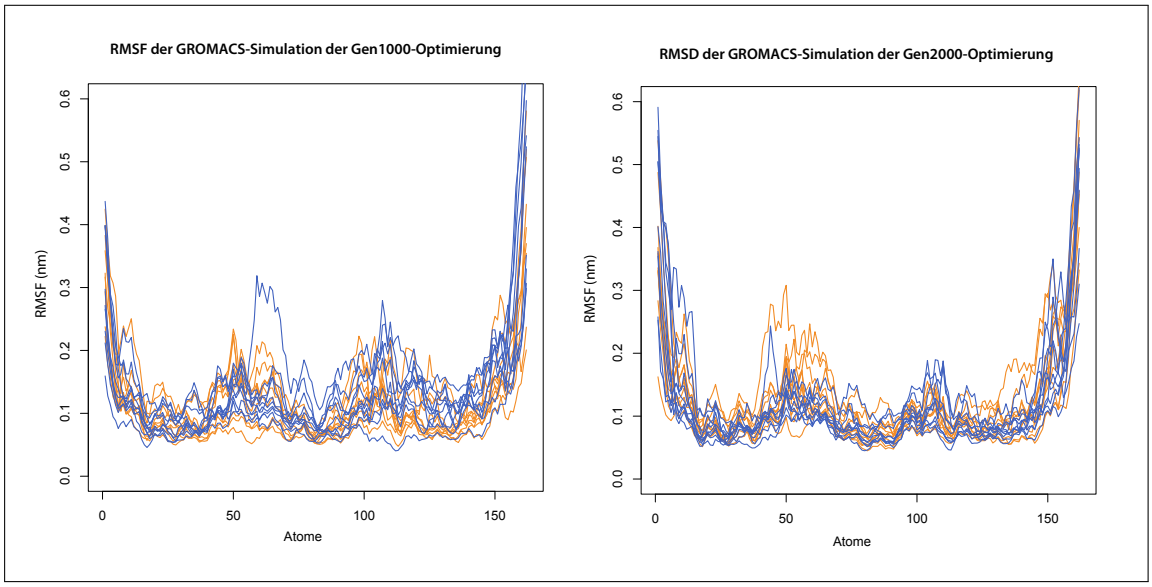


Abbildung 2.13: RMSF der GROMACS-Simulationen berechnet mit dem Programm `g_rmsf` aus dem GROMACS-Paket [39] wie in Absatz 2.5.2 beschrieben. In orange die Strukturen, die als beste von ERIS bewertet wurden, in blau die am schlechtesten bewerteten. Auf der horizontalen Achse sind die Atome der Struktur von N-Terminus nach C-Terminus und auf der vertikalen ist der RMSF in nm aufgetragen.

turen, die besser von ERIS bewertet wurden, insgesamt stabiler über ihre Simulationstrajektorie hinweg. Die insgesamt 40 Simulationen lassen sich an Hand von zwei Kriterien in vier Gruppen zu je 10 Simulationen aufteilen: Ein Kriterium ist die Anzahl der Generationen, das andere die Bewertung der Stabilität durch ERIS. Nach Begutachtung der Simulationstrajektorien wurden für jede dieser vier Gruppen vier Kandidaten ausgewählt, die einen möglichst guten Überblick über die strukturellen Veränderungen der Modelle aus dieser Simulationsgruppe geben. Die Strukturen des jeweils letzten Simulationsschrittes zeigt Abbildung 2.14. Zur Referenz wurden die Strukturen mittels Modeller 9v8 [71] an der Zielstruktur der Z-Domäne (PDB-ID: `1LP1`) ausgerichtet.

Bei der Betrachtung der Strukturen werden bei der Gen1000-Optimierung häufiger starke Deplatzierungen der ersten oder zweiten Helix beobachtet, zum Beispiel JW2 und JW55. Diese beiden Helices zeigen sich während der Gen2000-Simulationen stabiler. Ebenfalls kam es in der Gen1000-Optimierung zu einer teilweisen Entfaltung der dritten Helix bei JW26, die in den Gen2000-Simulationen nicht beobachtet wurde.

Wesentlich häufiger wurden Veränderungen der Position in der dritten Helix beobachtet. Extreme Verschiebungen der Helix treten jedoch nur bei den von ERIS am schlechtesten bewerteten Strukturen auf, etwa JW18, JW21 und JW55. Deplatzierun-

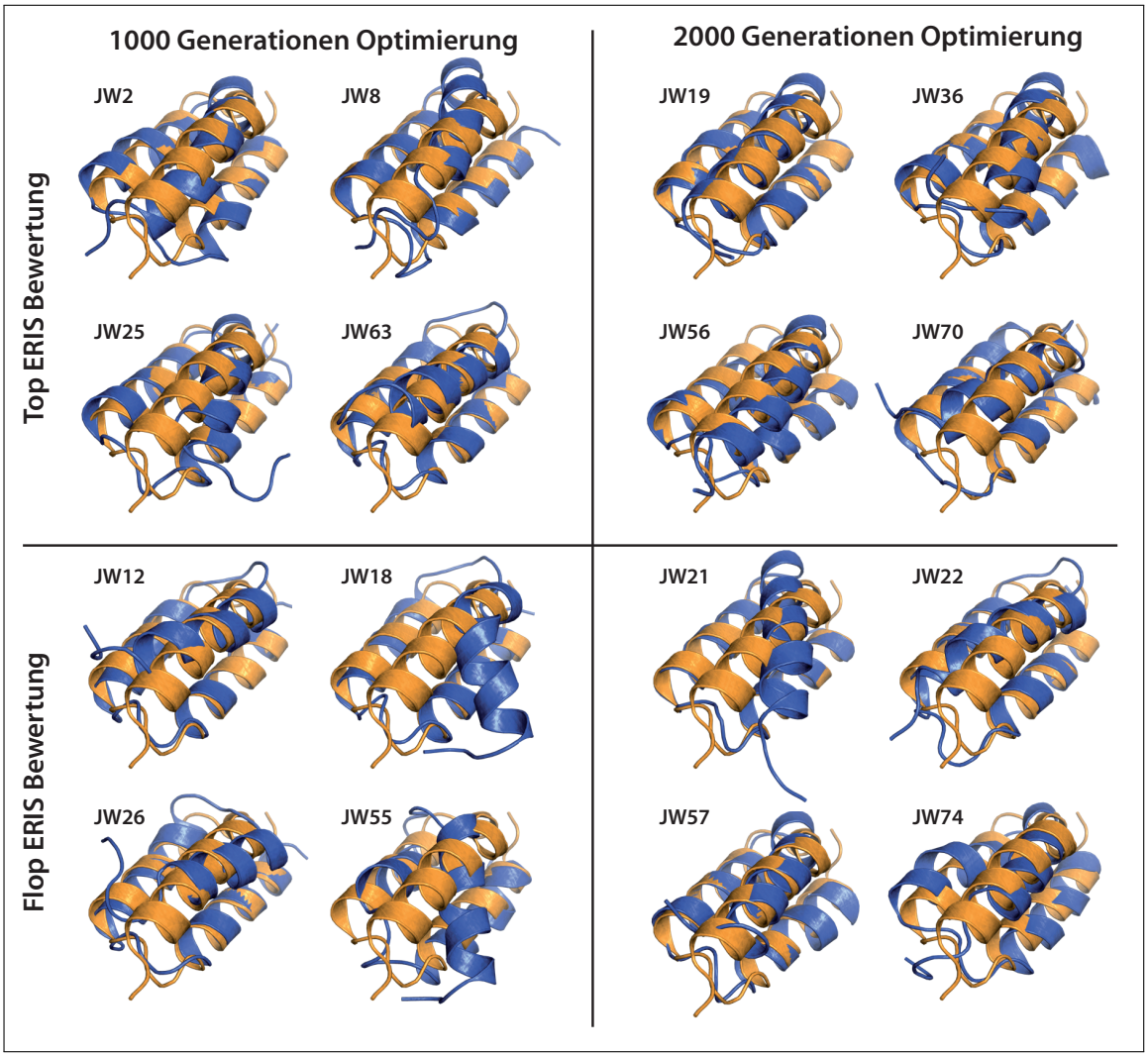


Abbildung 2.14: Strukturvergleich von je vier repräsentativen Strukturen der vier Simulationsgruppen (blau) ausgerichtet an der Zielstruktur der Z-Domäne (PDB-ID: `1LP1`) (orange). Zur zukünftigen Identifikation wurden die Strukturen mit Bezeichnern versehen (links oben an jeder Struktur). Die dritte Helix der Strukturen befindet sich in den Abbildungen im Vordergrund, der N-Terminus somit oben rechts.

gen treten ebenfalls bei den am besten bewerteten Strukturen auf (JW8, JW25 und JW36), werden aber weniger häufig beobachtet und sind nicht so stark ausgeprägt wie bei den am schlechtesten bewerteten Strukturen.

Die Simulation der Startsequenz modelliert auf die Struktur der Z-Domäne zeigt ebenfalls Verschiebungen der Helices zueinander. Abbildung 2.15 zeigt wie in Abbildung 2.14 die Struktur des letzten Simulationsschrittes ausgerichtet an der Z-Domäne. Neben einer Drehung der dritten Helix über Helix 1 und 2 zeigt sich eine Drehung der ersten Helix gegen Helix 2. Die neu eingebrachte Sequenz führt folglich zu einer De-

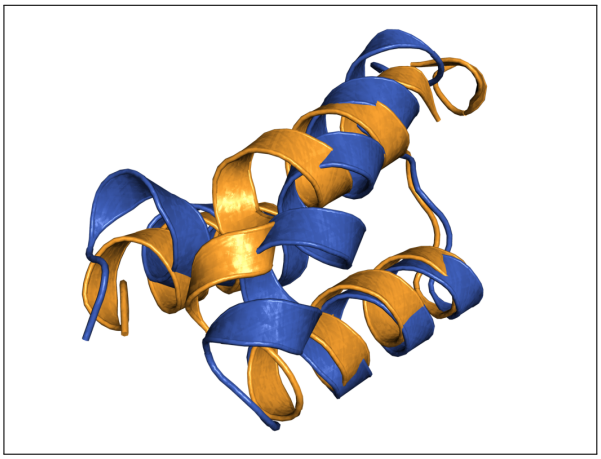


Abbildung 2.15: Startsequenz der GA-Optimierung modelliert auf die Struktur der Z-Domäne nach einer 20 ns langen GROMACS-Simulation (blau). Die Struktur wurde ausgerichtet an der Struktur der Z-Domäne (PDB-ID: `1LP1`) (orange).

Struktur	$k_{on}(M^{-1}s^{-1})$	Nettoladung	k_{on}^{rel}(Wildtyp)
DNA Bindehelix (Wildtyp)	$4{,}59 \cdot 10^8$	+5	1,000
Z-Domäne (Negativkontrolle)	0	-2	0,000
Startstruktur	$1{,}04 \cdot 10^8$	+5	0,227
JW19	$1{,}39 \cdot 10^7$	+3	0,030
JW56	$3{,}91 \cdot 10^7$	+5	0,085
JW70	$4{,}21 \cdot 10^8$	+5	0,917

Tabelle 2.2: Simulationsergebnisse der BrownDye-Simulation.

stabilisierung des gesamten Komplexes. Es ist nicht ausgeschlossen, dass die Funktion der Fc-Bindestelle durch die Drehung der ersten zur zweiten Helix verloren gegangen ist. Der große Abstand des N-Terminalen Endes von Helix 3 zum Rest des Proteins deutet auf einen fehlenden hydrophoben Kern im Protein hin.

2.7.4 BrownDye-Simulation der ausgewählten Strukturen

Auf Grund der besseren Ergebnisse in der Gen2000-Optimierung wurden drei Strukturen als Ergebnis des erweiterten Optimierungsprozesses gewählt. Für die ausgewählten Strukturen wurden wie in Absatz 2.6.1 beschrieben Brownian Dynamics-Simulationen mittels BrownDye [43] angefertigt. Die Ergebnisse sind in Tabelle 2.2 dargestellt. Aus den absoluten k_{on} Werten wurden auf Basis des Wildtypes, also der Simulation mit der DNA-bindenen Helix aus MyoD, die relativen Assoziationswerte für die übrigen Simulationen berechnet.

Für die MyoD DNA-Bindehelix ergab sich aus der Simulation eine Assoziationsrate von $4{,}59 \cdot 10^8 M^{-1}s^{-1}$. Die Simulation der Z-Domäne als Negativkontrolle zeigte wie erwartet keine Assoziation zur DNA über die Simulationszeit von 25.000 Einzelsimulationen. Der Grund dafür ist das Fehlen einer DNA-Bindestelle. Hinzukommend besitzt die Z-Domäne insgesamt eine negative Nettoladung und wird so von der ebenfalls negativ geladenen DNA abgestoßen. Die restlichen Modelle zeigten alle eine Assoziation zur DNA, jedoch unterscheiden sich die Ergebnisse stark. Die Startstruktur, die die unveränderte DNA-Bindestelle aus MyoD enthält, zeigte eine relative Assoziationsrate von 23%, obwohl die Nettoladung mit der des Wildtyps identisch ist. Von den drei Kandidaten aus dem Optimierungsprozess zeigte JW70 eine Assoziationsrate von $4{,}21 \cdot 10^8 M^{-1}s^{-1}$ und ist mit 92% auf Höhe des Wildtypes. Die anderen beiden liegen unter den Assoziationsraten der Startsequenz, zeigen also keine Verbesserung durch die Optimierung.

Bei Betrachtung der Nettoladung scheint ein gewisser Zusammenhang zwischen der Nettoladung des Proteins und der Assoziationsrate zu bestehen. Jedoch wird aus den Ergebnissen der Simulationen ebenfalls klar, dass die Nettoladung nicht der einzige Grund für einen hohen k_{on} Wert ist. Auf Grund der hohen Stabilität während der MD-Simulation und der guten Ergebnisse in der BrownDye-Simulation, wird JW70 für die Simulation im Komplex mit AMBER gewählt.

2.7.5 AMBER-Simulation von JW70

Wie in Absatz 2.6.2 beschrieben wurde JW70 im Komplex mit DNA 10 ns simuliert. Als Vergleich dienten Simulationen von der Startsequenz, der Z-Domäne und der Bindehelix von MyoD. Abbildung 2.16 zeigt die Strukturen nach der Simulation. Repositioniert wurde an Hand der DNA, so dass die DNA in der Abbildung repräsentativ für die DNA-Stränge der einzelnen Simulationen positioniert ist.

Die Ergebnisse der AMBER-Simulation decken sich mit den Ergebnissen der GROMACS und BrownDye-Simulationen. Die Z-Domäne zeigt keine DNA-Bindung und wandert von der initialen Position zu Beginn der Simulation weg von der DNA. Die verbleibenden drei Strukturen zeigen alle eine konstante DNA-Bindung über die Simulationsdauer. Die DNA-Bindung der konstruierten Strukturen ist dabei mit der Bindung der MyoD-Bindehelix vergleichbar. Ebenfalls bestätigt wird die Instabilität der Startsequenz. Nach der GROMACS-Simulation wurde eine starke Depositionierung der Helices beobachtet. In dieser Simulation wird ein Entfalten der Proteinstruktur beobachtet. Die eingebrachte DNA-bindene Helix verweilt zwar in Bindung mit der DNA, die Helices 1 und 2 lösen sich jedoch von der dritten Helix und driften von

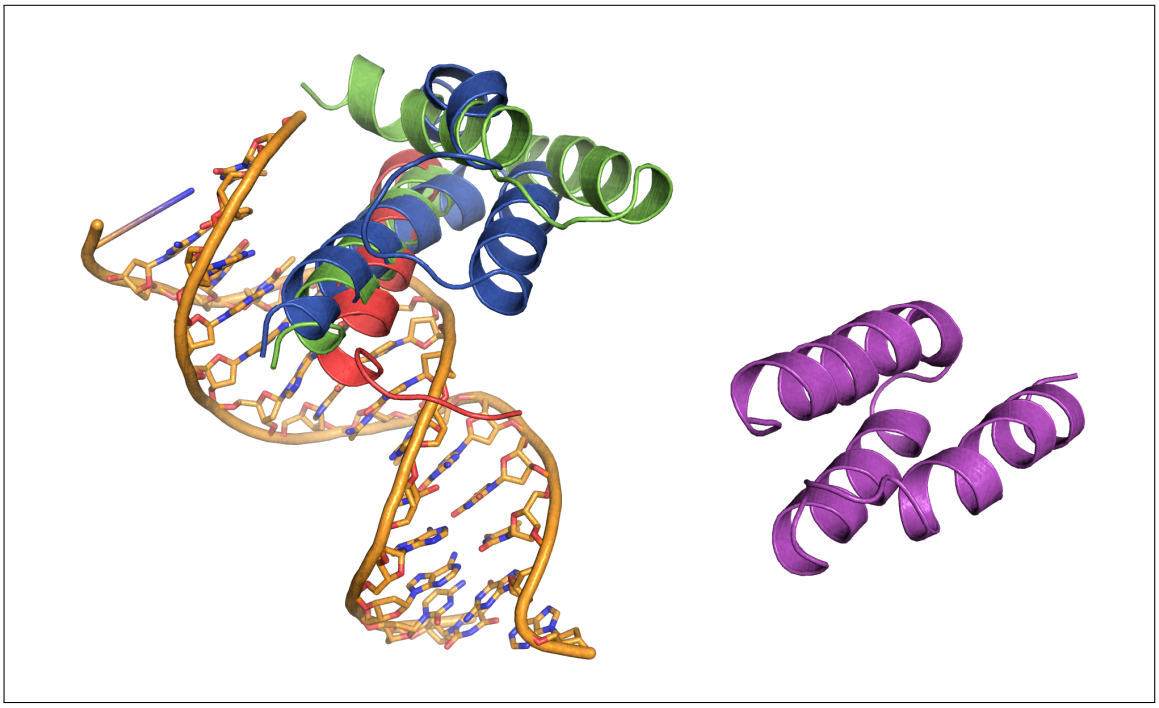


Abbildung 2.16: Ergebnis der vier AMBER-Simulationen. Ausgerichtet an der DNA (orange) wurden die vier Strukturen von JW70 (blau), der Startsequenz (grün), der Z-Domäne (violett) und der MyoD DNA-Bindehelix (rot).

der DNA weg ins Medium.

JW70 zeigt über die Simulation hinweg eine stabile Bindung zur DNA, bleibt strukturell jedoch nicht so stabil wie in der GROMACS-Simulation. Auffällig ist, dass Phe27 aus der Mitte der zweiten Helix über die Simulation hinweg in den hydrophoben Kern des Proteins wandert. Obwohl Phenylalanin eine hydrophobe Aminosäure ist, wurde sie über die GROMACS-Simulation hinweg zur Außenseite des Proteins hinein ins Medium platziert. Durch den engen Kontakt zur DNA wurde Phe27 nun zwischen der zweiten und dritten Helix hindurch in den Kern des Proteins verschoben. Der dicht gepackte Kern von JW70 wird somit aufgedrückt und die Helices 1 und 2 entfernen sich leicht von Helix 3. Trotzdem bleibt der hydrophobe Bereich zwischen den Helices vorhanden und stabilisiert diese zueinander über die Simulationsdauer hinweg.

2.8 Diskussion

Die Ergebnisse haben gezeigt, dass es möglich ist, mit Hilfe von Optimierungen durch einen Genetischen Algorithmus auf Basis einfacher Fitnessfunktionen eine aus zwei Proteinen zusammengesetzte Aminosäurensequenz zu optimieren, so dass dieses die

vorgegebene Struktur besser hält, als die nicht optimierte Sequenz. Verschiedene, etablierte Verfahren zeigen eine erhöhte Stabilität und Funktionalität im Vergleich zur Ausgangssequenz.

Duch Betrachtung der Sequenzen der Gen1000 und Gen2000 Generation fällt auf, dass die Sequenzen eines Optimierungslaufes sich jeweils in einer Gruppe befinden. Sie sind sich zueinander also ähnlicher als zu den Individuen der anderen Gruppe. Im Hinblick auf die sehr komplexe Fehleroberfläche des gestellten Problems ist dies nicht verwunderlich. Die Optimierungen konvergieren demnach nicht zu ein und dem selben Ziel. Durch Ausweiten des Suchraums ist es möglich, den Abstand dieser gefundenen Minima zu verringern. Eine Möglichkeit wäre, einen weniger elitären Selektionsoperator zu wählen oder die Anzahl der Individuen pro Generation zu erhöhen.

Ebenfalls auffällig bei Betrachtung des Konvergenzverhaltens der GA-Optimierungen war die unterschiedlich schnelle Konvergenz. Je nach Problem oder von der Optimierung gewähltem Lösungsweg ändert sich das Konvergenzverhalten. Somit wird eine Methode nötig, die eine Aussage darüber trifft, ob die Optimierung konvergiert ist oder nicht.

Der erste Schritt nach Absolvieren der Optimierung im GA ist das Erstellen der Strukturen aus den Ergebnissequenzen des GAs und die Bewertung genau dieser durch ERIS. Die Betrachtung der Ergebnisse von ERIS ließ vermuten, dass die Strukturen der Gen1000 während der GROMACS-Simulation insgesamt stabiler sind als die der Gen2000. Interessanterweise wurde genau diese Annahme von der MD-Simulation nicht bestätigt, sogar das Gegenteil war der Fall. Natürlich entsprechen die Ergebnisse der GROMACS-Simulation nicht exakt dem typischen Verhalten der Proteine. Durch die große Anzahl der Simulationen lassen sich jedoch die Ergebnisse der beiden Optimierungen untereinander vergleichen. Obgleich ERIS die Energiedifferenz zu der Struktur der Z-Domäne erstellt, kann die Schlussfolgerung gezogen werden, dass die absoluten Werte bei der Komplexität dieses Problems keine Aussagekraft besitzen. Da sich die Sequenzen der Optimierungsläufe untereinander jedoch sehr ähnlich sind, besitzt die Reihenfolge der Sequenzen sehr wohl Aussagekraft, was durch die GROMACS-Simulationen gezeigt wurde.

Mit den GROMACS-Simulationen sollte eine Aussage über die Stabilität der Sequenz beziehungsweise derer Struktur getroffen werden. Eine statistische Auswertung über den RMSD und den RMSF lieferte Hinweise auf strukturelle Eigenschaften eines gesamten Simulationssatzes. Eine angeschlossene manuelle Begutachtung der Simulationstrajektorie jeder Simulation bestätigte die Hinweise und bot genaueren Einblick in die strukturellen Veränderungen.

Das Testen durch BrownDye- und AMBER-Simulationen hat die Stabilität des aus-

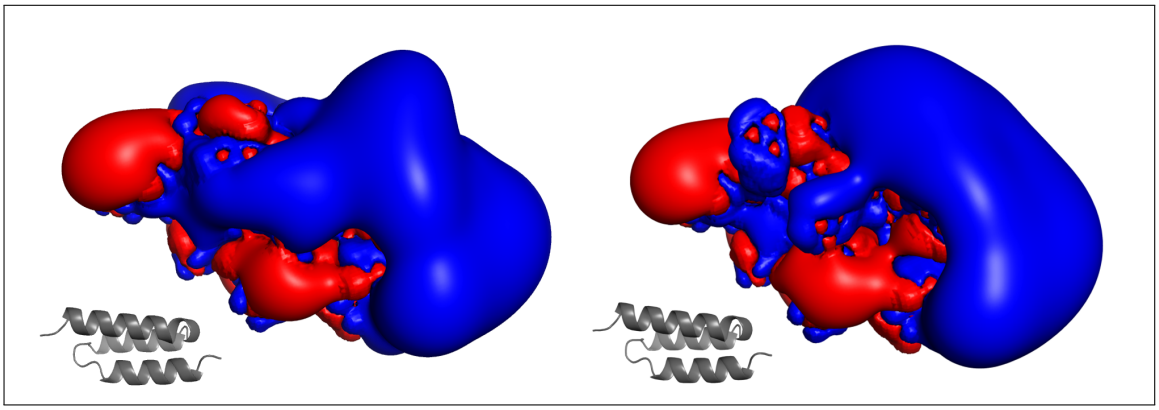


Abbildung 2.17: Elektrostatische Isoflächen von JW56 (links) und JW70 (rechts) bei 1 kT/e. Die Oberflächen wurden erstellt mit APBS [5]. Die Oberfläche der positiven Ladung ist blau gezeichnet, die der negativen Ladung rot. Links unter den jeweiligen Abbildungen sind die zugehörigen Proteine in Sekundärstrukturdarstellung in gleicher Orientierung abgebildet. Die dritte DNA-bindene Helix befindet sich oben.

gewählten Proteins bestätigt und Indizien geliefert, dass die neu konstruierte Bindestelle ihre Funktion wie erwartet erfüllt. Die Brownian Dynamics-Simulationen zeigten dabei auf, dass besonders die Elektrostatik für dieses Problem eine Rolle spielt. Insgesamt war der Trend zu erkennen, dass eine positivere Nettoladung des Proteins die k_{on} Werte der Proteine verbessert. Jedoch war zu beobachten, dass auch mit gleicher Nettoladung dieser Wert sehr schwanken kann.

Abbildung 2.17 zeigt eine Gegenüberstellung der elektrostatischen Isoflächen von JW70 und JW56 bei 1 kT/e, was in etwa 26,7 mV entspricht. Diese Darstellung macht die elektrostatischen Unterschiede im Bereich der dritten Helix des Proteins deutlich. Die Unterschiede können damit begründet werden, dass JW70 Assoziationsraten nahe des Wildtyps erreicht hat, JW56 hingegen nicht: JW70 zeigt eine stärkere positive Ladung im Bereich der dritten DNA-bindenen Helix.

Die Verteilung des elektrostatischen Potentials auf der Oberfläche des Proteins spielt bei diesem Problem also eine entscheidende Rolle. Somit sollte eine Fitnessfunktion die Elektrostatik abdecken, um diese schon während der Optimierung durch den GA zu optimieren.

Für JW70 zeigt die AMBER-Simulation eine mit dem Wildtyp vergleichbare DNA-Bindung. Die Instabilität der Startsequenz wurde auch während dieser Simulation bestätigt: Über die Simulationszeit hinweg löste sich die dritte Helix von den Helices 1 und 2 und das Protein begann sich zu entfalten, ein starker Hinweis auf einen nicht korrekt ausgebildeten hydrophoben Kern. Schließlich wurde dieser auch nicht optimiert.

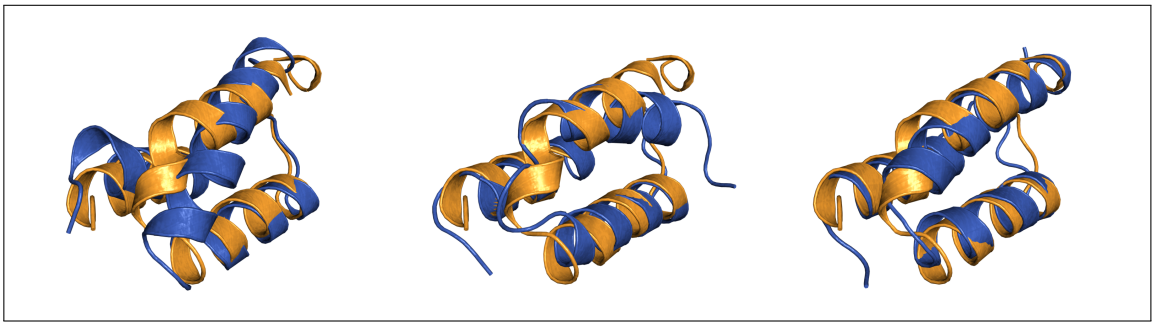


Abbildung 2.18: Vergleich verschiedener Simulationsergebnisse (blau) mit der Z-Domänen-Struktur nach 10 ns MD-Simulation (orange). Von links: Die Startsequenz des GAs, JW70 und JW19.

Eine Gegenüberstellung verschiedener Strukturen in Abbildung 2.18 zeigt das Potential der Methode. Trotz der verringerten Vorhersagegenauigkeit der Sekundärstrukturvorhersage und der Abstraktion von der Sequenz durch Nutzen eines *sliding-window*-Verfahrens für die Vorhersage der Hydrophobizität und des Molekulargewichtes wurden insgesamt durch das Verfahren gute Ergebnisse geliefert. Dies ist ebenfalls auf die große Anzahl an Individuen und Generationen zurückzuführen, die erst durch starke Vereinfachung der Fitnessfunktionen möglich ist.

Die Startsequenz wurde so optimiert, dass die erwünschte Struktur komplett erhalten bleibt, wie es zum Beispiel bei JW19 beobachtet werden kann. JW70 zeigt nur leichte Abweichungen von der Zielstruktur, erfüllt jedoch die Anforderungen der eingebrachten DNA-Bindestelle und erzielt zu MyoD vergleichbare DNA-Bindeeigenschaften.

Mit Hilfe der bisher aus diesem Kapitel gewonnenen Erkenntnissen sollen nun in Kapitel 3 verschiedene Verbesserungen am GA vorgenommen werden. Eine Konvergenzvorhersage soll Aussagen darüber liefern, ob die Optimierung konvergiert ist. Neben Parametern wie der Anzahl der Generationen, der Anzahl an Individuen in einer Population und der Mutationsrate sollen weitere genetische Operatoren darauf getestet werden, ob die Optimierungsleistung verbessert werden kann. Eine neue Fitnessfunktion wird der wichtigen Rolle der Elektrostatik gerecht.

3 Optimierung des Genetischen Algorithmus

«Does it worry you that you don't talk any kind of sense?»

Douglas Adams

3.1 Einleitung

Wie bei den meisten Optimierungsmethoden gibt es auch bei den Genetischen Algorithmen unzählige Stellschrauben, an denen gedreht werden kann, um die Leistung des Algorithmus zu verbessern. Dass das gestellte Problem mit dem GA bearbeitet werden kann, wurde in Kapitel 2 schon gezeigt. In diesem Kapitel geht es um das Einstellen der Stellschrauben, um eine konsistente und verlässliche Optimierung zu erhalten.

Grundlage für die folgenden Untersuchungen ist eine Konvergenzanalyse. Damit sollen Aussagen über das Laufzeitverhalten des GAs getroffen werden. Anschließend können erste einfache Parameter an dem GA geändert werden, zum Beispiel die Generationenzahl oder die Anzahl der Individuen. Sind hierfür korrekte Parameter gefunden, sollen die genetischen Operatoren untersucht werden. Vor allem für die Selektion existiert eine große Auswahl an unterschiedlichen Verfahren.

Bei den Brownian Dynamics-Untersuchungen im letzten Kapitel wurde die Bedeutung der Elektrostatik für eine erfolgreiche DNA-Bindung klar. Aus diesem Grund

soll die Elektrostatik mit in die Optimierung einfließen. Eine neue Fitnessfunktion berechnet auf der Tertiärstrukturebene durch einen Vergleich der elektrostatischen Hüllen der Individuen einen Fitnesswert. Der Nutzen dieser neuen Methode soll hier ebenfalls untersucht werden.

3.2 Konvergenzvorhersage

Zur Vorhersage der Konvergenz einer Optimierung im GA wird das Stoppkriterium der kleinsten Quadrate (LSSC) [36] aus der Software von Wagner et al. [90] genutzt. Dabei handelt es sich um eine Sammlung von Skripten für das Programm MATLAB, welches in Version R2011b [59] genutzt wird.

Das LSSC basiert auf drei Qualitätskriterien, für welche einzeln eine lineare Regression durchgeführt wird. Die Optimierung gilt als konvergiert, sobald die Steigung der Regressionsgeraden und die Abweichung der Punkte um die Regressionsgerade unter einen Schwellenwert fallen. Die einzelnen Elemente werden im Folgenden genauer behandelt.

Qualitätsindikatoren Da der mathematische Hintergrund der drei verwendeten Qualitätsindikatoren in der aufgeführten Fachliteratur hinreichend beschrieben ist, werden die Methoden hier lediglich umrissen. Die Qualitätsindikatoren arbeiten dabei binär; sie vergleichen eine Lösung mit einer Referenzlösung. Lösungen sind in diesem Fall die Fitnesswerte aller Individuen der aktuellen Generation. Die Indikatoren wurden ursprünglich entwickelt, um als Referenzlösung die optimale Pareto-Front des Problems zu setzen. In dem hier gegebenen Optimierungsproblem ist diese jedoch nicht bekannt. Da es sich aber um ein Minimierungsproblem handelt und das theoretische Minimum der Fitnesswerte 0 ist, wird als optimale Pareto-Front die Menge bestehend aus nur dem einem Vektor (0,0,0) für Optimierungen mit drei Fitnesswerten gewählt.

Der Hypervolumen-Indikator (HV) [96] $I_H(A)$ berechnet das Volumen des Raumes, welcher durch die Punkte aus der Menge A von Individuen einer Generation und einer Referenzmenge N aufgespannt wird. N ist dabei eine Menge von Punkten, die keine anderen Punkte dominieren. Für den Indikator gilt dann:

$$I_{HV}(A,B) = \begin{cases} I_H(B) - I_H(A) & \text{wenn } \forall b \in B\, \exists a \in A \text{ mit } b \succ a, \\ I_H(A+B) - I_H(A) & \text{sonst.} \end{cases} \tag{3.1}$$

Der Epsilon-Indikator (EPS) [47, 97] basiert auf dem Dominanzkonzept. Verglichen werden zwei Mengen von Punkten A und B mit Hilfe eines Wertes ε. Für ein Minimierungsproblem bestimmt der Indikator das kleinste ε, das nötig ist, damit A die Menge B dominiert, wenn zu jedem Element aus B ε hinzuaddiert wird:

$$I_{EPS}(A,B) = \inf_{\varepsilon\in\mathbb{R}}(\forall b \in B\, \exists a \in A, \text{ so dass } a \succ b + \varepsilon) \tag{3.2}$$

Der Indikator der gegenseitigen Dominanzrate [58] (englisch: *mutual domination rate*, MDR) ist im engeren Sinne kein Qualitäts-, sondern ein Fortschrittsindikator. Er untersucht, wie sehr eine Generation von Individuen die Vorgängergeneration dominiert und umgekehrt. Gegenüber HV und EPS hat er den Vorteil, dass der Rechenaufwand für die Berechnung weit geringer ist und die Generation nicht zu einem Optimum hin verglichen wird. Um den Indikator zu definieren, wird die Funktion Φ genutzt, welche die Elemente von A zurück gibt, die von mindestens einem Element aus B dominiert werden:

$$C = \Phi(A,B) \Leftrightarrow \forall c \in C \text{ mit } c \in A\, \exists b \in B, \text{ so dass } b \succ c \tag{3.3}$$

Damit lässt sich nun der Indikator definieren durch:

$$I_{MDR}(\mathcal{P}_{t-1}, \mathcal{P}_t) = \frac{\|\Phi(\mathcal{P}_{t-1}, \mathcal{P}_t)\|}{\|\mathcal{P}_{t-1}\|} - \frac{\|\Phi(\mathcal{P}_t, \mathcal{P}_{t-1})\|}{\|\mathcal{P}_t\|} \tag{3.4}$$

mit den Pareto-Fronten $\mathcal{P}$ zu den Zeiten t und $t-1$, also der aktuellen Generation und der Vorgängergeneration. Der Indikator gibt Werte aus dem Bereich $[-1,1]$ zurück. $I_{MDR} = 1$ steht dabei für den Fall, dass sich alle Individuen der Pareto-Front verbessert haben. Bei $I_{MDR} = 0$ gibt es keine Änderungen und $I_{MDR} = -1$ beschreibt den Fall, dass sich die Pareto-Front der aktuellen Generation zur Vorgängergeneration in allen Elementen verschlechtert hat.

Konvergenzkriterium Diese drei Indikatoren werden nun als Basis für eine lineare Regression der Form $y = bx + a$ hergenommen. Mit Hilfe der Methode der kleinsten Quadrate kann für eine gegebene Fensterbreite w die Regression gelöst und somit die Werte für a und b berechnet werden. Anschließend wird die Abweichung φ der einzelnen Werte von der Regessionsgeraden berechnet. Durch die Methode der kleinsten Quadrate entsprechen die Abweichungen einer χ^2-Verteilung:

$$\varphi = \frac{\sum_i (y_i - (bx_i + a))^2}{w} \approx \frac{\chi^2}{w} \tag{3.5}$$

Die Varianz und der Erwartungswert der χ^2-Verteilung sind bekannt mit $\mu(\chi^2_n) = n$

respektive $\sigma^2(\chi_n^2) = 2n$. Mit Hilfe der Tschebyscheff-Ungleichung lässt sich nun ein Schwellenwert τ abschätzen, so dass 99% der Individuen einer χ^2-Verteilung darunter fallen:

$$\tau = \mu + 3\sigma = 1 - \frac{2}{w} + 3 \cdot \sqrt{\frac{2}{w} - \frac{4}{w^2}} \tag{3.6}$$

Fällt nun also der Wert φ aus Formel 3.5 unter den Wert des Schwellenwertes τ aus Formel 3.6, kann die Verteilung der Indikatoren der letzten w Werte als gleichmäßig angesehen werden. Zusätzlich soll die Steigung der Regressionsgeraden b unter einen Wert b_{target} fallen, der für das gegebene Problem gesucht werden muss. Sind diese beiden Kriterien erfüllt, so ist der GA unter diesen Kriterien konvergiert [36].

Je nach Parametrisierung des GAs ist der Schwellenwert für die Abweichung der einzelnen Werte von der Regressionsgeraden unterschiedlich zu wählen. Ein elitärer Ansatz erzeugt weniger Schwankungen in den Fitnesswerten der Individuen als ein Ansatz, bei dem auch alte, nicht favourisierte Individuen mit in die neue Generation übernommen werden. Als Fensterbreite werden für die folgenden Untersuchungen jeweils die letzten 250 Schritte berücksichtigt.

3.3 Optimierung der GA-Parameter

Die Parametrisierung eines Genetischen Algorithmus ist stark von dem gestellten Optimierungsproblem abhängig. Für die Anzahl der Individuen finden sich etwa Werte von 15 [34] bis 100.000 [55]. Es ist sogar offensichtlich, dass eine unendlich große Menge von Individuen direkt zur optimalen Lösung unter den gegebenen Fitnessfunktionen führt. Auch bei der Generationenzahl findet man unterschiedlichste Werte von bis zu 40.000 Generationen und mehr [60]. Stets muss die Populationsgröße und die Generationenzahl zur Laufzeit des Algorithmus abgewogen werden.

Um die zur Verfügung stehende Rechenleistung möglichst gut auszunutzen, soll für dieses Problem auf Basis der Ergebnisse aus dem ersten Teil ein neuer Satz von Parametern gesucht werden, der möglichst gut auf das hier gestellte Problem angepasst ist. Genauer beleuchtet wird die Anzahl der Generationen, die Anzahl der Individuen in einer Population und die Mutationsrate.

Die Optimierung aller Parameter gleichzeitig setzt eine große Anzahl an Testläufen des GAs voraus. Da bei einer Laufzeit von 2-3 Tagen auf einem Cluster von 30 Computern pro Optimierung eine breite Parametersuche nicht praktikabel ist, wurden folgende Parameter zur Überprüfung gewählt: Die Mutationsrate wird mit den Werten 0,005, 0,01 und 0,02 gewählt. Eine Mutationswahrscheinlichkeit von 0,02 pro

Aminosäure enspricht bei der verwendeten Sequenz mit 54 Aminosäuren im Mittel einer Mutation pro Individuum je Generation und soll bei diesen Versuchen das Maximum darstellen. Die Werte 0,01 und 0,005 führen weniger Mutationen ein und wirken damit konservierender auf die rekombinierten Individuen.

Für die Anzahl an Individuen hat sich 600 in dem vergangenen Experiment als geeignet gezeigt. Eine Verdopplung der Individuen würde zu eine Verdopplung der Laufzeit führen und wird darum nicht mehr untersucht. 300 Individuen zu berechnen halbiert die Laufzeit entsprechend und soll ebenfalls untersucht werden. Bleibt dabei die Optimierungsleistung des Algorithmus gleich, wäre dies eine gute Alternative zu 600 Individuen.

Werte für die Anzahl der Generationen brauchen nicht einzeln untersucht werden. Aus der Kombination von drei verschiedenen Mutationsraten und zwei Populationsgrößen ergeben sich sechs Simulationen, die jeweils über 2000 Generationen simuliert werden. Mehr als 2000 Generationen sollten nicht nötig sein, wie schon die Ergebnisse aus Kapitel 2 gezeigt haben. Mit Hilfe der Konvergenzvorhersage lässt sich jedoch nach der Optimierung eine ausreichende Zahl von Iterationen ermitteln, die dann für spätere Simulationen verwendet wird.

Es darf nicht vergessen werden, dass es sich bei dieser Optimierung durch einen GA um einen stochastischen Optimierungsprozess handelt. Die Ergebnisse können also nur zufällig besonders gut oder besonders schlecht ausfallen. Im Folgenden werden die Optimierungen stets dreimal initialisiert und durchgeführt. Der Zeitaufwand bleibt vertretbar, einzelne zufällige Ausreißer sind jedoch leichter zu identifizieren und die Ergebnisse werden verlässlicher.

3.4 Optimierung der GA-Operatoren

Nachdem die Parameter des GAs optimiert wurden und ein guter Kompromiss zwischen Laufzeit und Optimierungsleistung gefunden wurde, sollen nun verschiedene genetische Operatoren untersucht werden. Bei den Operatoren zur Mutation und Rekombination gibt es zwar einige Varianten der verwendeten Versionen, diese machen meist jedoch nur bei speziellen Problemstellungen Sinn, welche bestimmte Restriktionen in den Operatoren voraussetzen. Für die in dieser Arbeit gestellten Aufgaben soll weiter mit den klassischen Operatoren auf Zufallsbasis gearbeitet werden.

Mehr relevante Operatoren existieren für die Selektion. Hier reicht die Bandbreite von elitären bis zu zufälligen Auswahlverfahren. Dabei sind die Übergänge je nach Verfahren fließend. Elitäre Ansätze führen zu einer raschen Konvergenz, gelangen dabei gerade bei komplexen Problemen oft in lokale Minima. Sie eignen sich vor allem für

Probleme mit einer flachen Fehleroberfläche, für die auch Gradientenabstiegsverfahren gute Ergebnisse liefern können. Wesentlich langsamer konvergieren Optimierungen mit Selektionsoperatoren, die auch schwache Individuen berücksichtigen. Der Grad dieser Berücksichtigung bestimmt dabei die Konvergenzgeschwindigkeit. Auf der anderen Seite wird der Suchraum besser durchsucht. Es ist somit wahrscheinlicher, in das globale Minimum zu gelangen.

Die Konvergenzgrafiken aus Absatz 2.7.1 zeigen die typischen Vor- und Nachteile elitärer Ansätze: Die Optimierungen konvergieren nach kurzer Zeit und zeigen kaum noch Veränderungen. Jedoch fällt schon bei der Betrachtung von zwei Läufen das unterschiedliche Niveau der Fitnesswerte auf. So ist alleine durch Betrachtung der Fitnesswerte klar, dass die Optimierungen in beiden Fällen nicht dasselbe Ziel erreicht haben.

Da in dem GA an zwei Stellen in einer Iteration ein Selektionsoperator verwendet wird, müssen diese beiden Operatoren sinnvoll aufeinander abgestimmt werden. Ein Blick auf Abbildung 2.4 aus Absatz 2.3 soll dabei helfen, die Wirkung einer Operatorkombination besser abzuschätzen. Es sollen folgende Kombinationen genauer untersucht werden:

- **Roulette-Selektion** und **Beste-Individuen** (BCP)
  Diese Kombination wurde bisher verwendet. Der Elternpool für die Rekombination wird durch eine Roulette-Selektion erzeugt. Nach der Mutation werden die besten Individuen aus der alten und der Nachkommengeneration für die neue Population gewählt. Dies ist die in den gängigen Ansätzen verwendete Selektionsart.

- **Roulette-Selektion** und **nur Kindergeneration** (CCP)
  Zum Erstellen des Elternpools wird die Roulette-Selektion verwendet. In die Nachfolgegeneration hingegen gelangen nur die rekombinierten Individuen. Individuen der vorherigen Generation gehen bei dieser Art der Selektion verloren. Diese Kombination von Selektionsoperatoren übt nur noch einen sehr geringen Selektionsdruck aus. Werden zwei sehr gute Individuen rekombiniert und anschließend eine Mutation eingeführt, kann das resultierende Individuum eine sehr schlechte Fitness aufweisen und trotzdem in die Nachfolgegeneration gelangen. Diese Art der Selektion kommt der natürlichen am nächsten.

- **Zufällige Selektion** und **Roulette-Selektion** (PCP)
  Bei dieser Kombination werden die Eltern für die Rekombination zufällig gewählt. Der Selektionsdruck wird nur noch bei der Auswahl der Individuen für

die Folgegeneration ausgeübt. Hier wird die Roulette-Selektion angewandt, um neben vielen guten Individuen auch einige weniger gute mit in die Folgegeneration zu nehmen. Es wird ein Kompromiss zwischen den beiden vorherigen, extremen Ansätzen gebildet.

Bei den nicht elitären Ansätzen kommt es durch die Mitnahme von schlechten und dem eventuellen Verlust von sehr guten Individuen zu Schwankungen in den Fitnesswerten, auch wenn der Algorithmus schon konvergiert ist. Um diese Schwankungen zu quantifizieren, wird wie schon bei den GROMACS-Simulationen der RMSF berechnet. In diesem Fall wird er von dem Mittel der Fitnesswerte aller Individuen einer Generation gebildet. Weil sich die Werte abgesehen von den Schwankungen, zu der Konvergenz hin entwickeln, kann der RMSF nicht an Hand nur eines Referenzpunktes ermittelt werden. Als Referenzpunkt wird laufend das Mittel aus den 100 Werten gebildet, die den zu berechnenden Punkt umgeben. Somit kann der RMSF für die ersten und letzten 49 Punkte nicht berechnet werden. Es wird also der RMSF der Fitnesswerte $s = (s_{50}, \ldots, s_N - 50)$ einer N-Generationen-Optimierung berechnet durch:

$$\mathrm{RMSF}(s) = \sqrt{\frac{1}{t}\sum_{i=50}^{N-50}[s_i - \sigma_i(s)]^2} \quad \text{mit} \tag{3.7}$$

$$\sigma_i(s) = \frac{1}{100}\sum_{k=-49}^{50} s_{i-k}$$

Dabei ist $\sigma_i(s)$ der Referenzpunkt für den i-ten Wert.

Zusätzlich zu den genetischen Operatoren wird die Auswirkung des eingeschränkten Aminosäurenalphabets auf die Konvergenz und die Fitnesswerte untersucht. Bisher wurden nur die Aminosäuren für die Optimierung verwendet, die auch in der Startsequenz des GAs vorkommen. Nun soll jede der drei Konfigurationen einmal mit und einmal ohne eingeschränktes Aminosäurenalphabet untersucht werden. Es werden wie auch schon bei der Parameteroptimierung drei Optimierungen für einen Operatorensatz gestartet, um konstante Ergebnisse zu erhalten. Insgesamt werden demnach 18 Optimierungsläufe durchgeführt und ausgewertet.

3.5 Fitnessfunktion: Epitopsy

Bei der Durchsicht der Konvergenzdiagramme bisheriger GA Optimierungen fällt auf, dass sich die Fitnesswerte der Hydrophobizitäts- und Molekulargewichtsvorhersagen

sehr ähnlich verhalten. Beide basieren auf der Primärstruktur des Proteins, also direkt auf der Sequenz, und sind dadurch methodisch sehr ähnlich. Um eine Komplexitätssteigerung der Optimierung durch Einführung einer vierten Fitnessfunktion zu vermeiden, stellt sich die Frage, ob eine der beiden Fitnessfunktionen durch eine neue ersetzt werden kann.

Da schon Funktionen auf Basis der Primär- und Sekundärstruktur vorhanden sind, soll untersucht werden, welche Verbesserungen eine Fitnessfunktion auf Basis der Tertiärstruktur des Proteins macht. Als Tertiärstruktur bezeichnet man die gesamte räumliche Struktur eines Proteins.

Es hat sich in der Vergangenheit gezeigt, dass die Elektrostatik eine wichtige Eigenschaft von Proteinen ist und vor allem für die Interaktion von Proteinen zueinander von relevanter Bedeutung ist [23, 26]. Wie in Absatz 2.6.1 gesehen, spielt die Elektrostatik auch für das hier behandelte Optimierungsproblem eine Rolle. Aus diesem Grund wird ein Vergleich der elektrostatischen Hüllen als Fitnessfunktion eingeführt. Sie wird die Molekulargewichtsvorhersage ersetzen, da die Hydrophobizität von höherer Relevanz für das gestellte Problem ist.

Aus der Poisson-Gleichung geht hervor, dass das elektrostatische Potential $\varphi(r)$ für heterogene Medien an der Position r direkt abhängt von der Ladungsdichte ρ und der dielektrischen Leitfähigkeit des Mediums ε, auch Permittivität genannt:

$$\Delta\varphi(r) = -\frac{\rho}{\varepsilon}. \tag{3.8}$$

Dabei ist Δ der Laplace-Operator und definiert als $\Delta\varphi(r) = \nabla \cdot (\nabla\varphi(r))$ mit der Divergenz $(\nabla\cdot)$ und dem Gradienten (∇). Wird der Gradient auf ein Skalarfeld angewandt, so ist das Ergebnis ein Vektorfeld, welches für jeden Skalar die Richtung der Änderung angibt. Die Divergenz angewandt auf ein Vektorfeld ergibt ein Skalarfeld. Für ein elektrisches Feld gibt dieses für jeden Punkt an, in welchem Maße das Feld zu dem Punkt hin oder von dem Punkt weg gerichtet ist.

Die Permittivität ist eine Eigenschaft von Materialien, die mit dem elektrostatischen Feld wechselwirken. So wird die elektrische Feldkonstante ε_0 des Vakuums mit einem Faktor $\varepsilon(r)$ für das wechselwirkende Material an der Stelle r multipliziert: $\varepsilon = \varepsilon_0\varepsilon(r)$. Typisch für Wasser ist ein Wert von 78-80, für Protein etwa 2-4. Damit lässt sich die Poisson-Gleichung umstellen zu

$$\varepsilon_0\nabla \cdot [\varepsilon(r)\nabla\varphi(r)] = -\rho. \tag{3.9}$$

Nun werden noch die Effekte von freie Ionen I an Position r zu der Lösung hinzugefügt:

$$\varepsilon_0 \nabla \cdot [\varepsilon(r) \nabla \varphi(r)] = -\rho - \sum_i I_i(r) \tag{3.10}$$

Um die Poisson-Gleichung zu lösen und das elektrostatische Potential im Punkt r zu berechnen, muss die Verteilung der Ladungsdichte der Ionen $\sum_i I_i(r)$ bekannt sein. Für kanonische Ensembles, also Systeme mit fester Teilchenzahl und Temperatur, folgt die Ladungsdichte einer Boltzmann-Verteilung. Positiv geladene freie Ionen werden nun von negativen Ladungen angezogen und umgekehrt. Durch die Boltzmann-Gleichung gilt dann:

$$I_i(r) = c_i^{\infty} z_i q \lambda(r) e^{\frac{-z_i q \varphi(r)}{k_B T}} \tag{3.11}$$

mit c^{∞} der Konzentration des Ions, z_i der Ladung des Ions, q der Ladung eines Protons, $\lambda(r)$ ein Maß für die Zugänglichkeit des Ortes r zu den Ionen in der Lösung, k_B der Boltzmannkonstante und T der Temperatur. Durch Einsetzen ergibt sich nun die Poisson-Boltzmann-Gleichung mit

$$\varepsilon_0 \nabla \cdot [\varepsilon(r) \nabla \varphi(r)] = -\rho(r) - \sum_i c_i^{\infty} z_i q \lambda(r) e^{\frac{-z_i q \varphi(r)}{k_B T}}, \tag{3.12}$$

die durch die Hilfe numerischer Verfahren linearisiert und gelöst werden kann.

Die Vorhersage der Elektrostatik für die Individuen basiert auf dem von Jan Nikolaj Dybowski entwickelten Programm Epitopsy [21] und gliedert sich in mehrere Schritte:

- Erstellen einer Gitterhülle um die Zielstruktur
- Berechnen der elektrostatischen Wechselwirkung an den Gitterpunkten
- Modellieren der Sequenz des Individuums auf die Zielstruktur
- Berechnen der elektrostatischen Wechselwirkung auf vorhandenem Gitter basierend auf der Sequenz des Individuums
- Vergleich und Bewertung der beiden elektrostatischen Hüllen

Erstellen einer Gitterhülle und Berechnen der Elektrostatik Die Elektrostatik wird für eine gegebene Struktur im PDB-Format berechnet. Diesem Format fehlen jedoch wichtige Parameter, die durch das Programm PDB2PQR [20] ergänzt werden. Ergänzt werden die Radien und die Ladungen zu jedem Atom in der Struktur, basierend auf dem CHARMM27 Kraftfeld [57]. Das Ergebnis wird als PQR-Datei gespeichert und für die nächsten Schritte verwendet.

Der *Adaptive Poisson-Boltzmann Solver* (APBS) in Version 1.3.0 [5] wird verwendet, um ein dreidimensionales Gitter über die Struktur zu legen und für jeden Gitterpunkt die Poisson-Boltzmann-Gleichung zu lösen, welche die Elektrostatik von Molekülen in Lösung beschreibt. Es wird ein kubisches Gitter verwendet, in dem die Gitterpunkte einen Abstand von 1 Å besitzen. Berechnet wird die Elektrostatik bei einer Temperatur von 310 K und einer einfachen Debye-Hückel-Kugel als Randbedingung. Die dielektrischen Konstanten wurden mit 4 für das Protein und 78 für die Lösung festgelegt.

Die Hülle um die Struktur soll einen Abstand von 6 Å zur Oberfläche des Proteins besitzen. Dies hat sich als guter Wert für den Vergleich der Elektrostatik verschiedener Strukturen gezeigt [22]. In einem ersten Schritt wird dazu die Van der Waals-Oberfläche der Struktur berechnet und dieser die jeweils nächsten Gitterpunkte zugeordnet. Die Berechnungen basieren auf den Werten, die von PDB2PQR in die pqr-Datei geschieben wurden. Anschließend wird diese Oberfläsche um 6 Å in Richtung des Mediums vergrößert. Dazu wird die Oberfläche schrittweise um einen Gitterpunkt nach außen verschoben. Durch sechsmaliges Ausführen wird eine Distanz von 6 Å zur Van der Waals Oberfläche des Proteins erreicht.

Bei diesem Verfahren liegt die vergrößerte Oberfläche weiterhin auf dem Gitter, welches APBS erzeugt hat. Somit können der Oberfläche die elektrostatischen Werte des bestehenden Gitters direkt zugeordnet werden. Abbildung 3.1 zeigt die ausgewählten Gitterpunkte zu der erzeugten Hülle um die Z-Domäne.

Nun liegt also eine Hülle mit Werten für die Elektrostatik um die Z-Domäne vor. Für diesen Optimierungsfall ist die Z-Domäne jedoch nicht das Optimierungsziel. Ziel der Optimierung soll sein, die Elektrostatik der DNA-Bindestelle über Helix 3 zu erhalten und die Elektrostatik des Fc-bindenen Teils der Z-Domäne über Helix 1 und 2 unverändert zu übernehmen. Aus diesem Grund wird die Elektrostatik für den Teil der Hülle um Helix 3 neu berechnet, die Form der Hülle bleibt jedoch erhalten. Dazu wird auf die Struktur der Z-Domäne die Startsequenz modelliert, welche die DNA-Bindestelle auf Helix 3 enthält. Für die ausgewählten Hüllenpunkte wird nun die Elektrostatik auf Basis der geänderten Struktur neu berechnet. Das Vorgehen und die Parameter dafür bleiben dabei wie oben beschrieben.

Ergebnis dieser Schritte ist also eine Punktwolke mit 6 Å Abstand zu der Van der Waals Oberfläche der Z-Domäne. Die Punktwolke enthält auf jedem Punkt das elektrostatische Potential, die das Zielprotein an dieser Stelle besitzen soll: Über Helix 1 und 2 die der ursprünglichen Fc-Bindestelle, über Helix 3 die der DNA-Bindestelle aus MyoD. Diese Punktwolke dient bei der Bewertung der Individuen in der Optimierung als Referenz. Die bisherigen Schritte müssen nur einmal vor der Optimierung

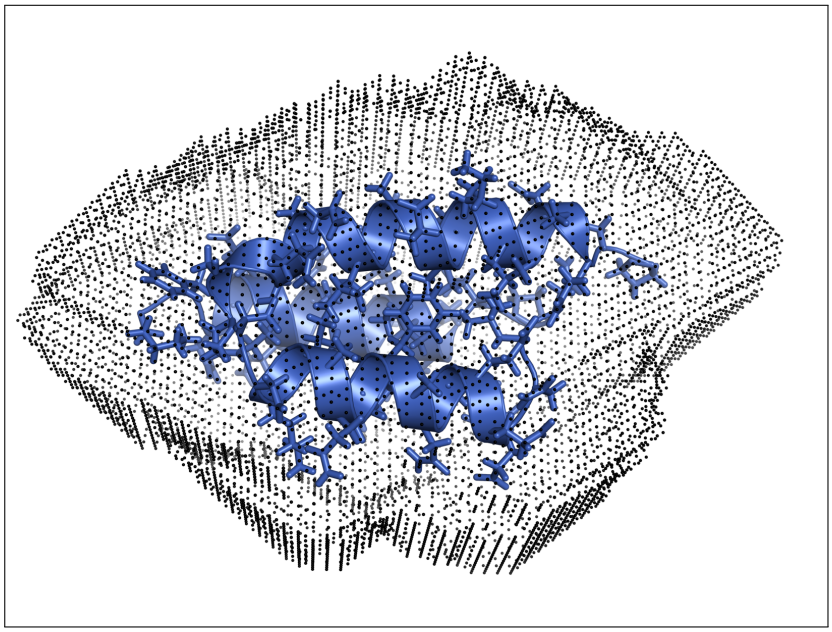


Abbildung 3.1: Punkte der Referenzhülle für die Elektrostatikberechnung in Epitopsy mit 6 Å Abstand zur Van der Waals Oberfläche. Die Hülle wurde um die Struktur der Z-Domäne erstellt (PDB-ID `1LP1`)

.

ausgeführt werden.

Modellieren der Struktur Zur Berechnung der Elektrostatik der Sequenz eines Individuums, das der Fitnessfunktion übergeben wird, wird eine Struktur benötigt. Hierfür wird die Struktur der Z-Domäne verwendet, von der auch die Hülle konstruiert wurde (PDB-ID: `1LP1`). Mit dem Programm *Modeller* [71] in der Version 9v8 wird dazu die übergebene Sequenz an der Sequenz der Z-Domäne ausgerichtet und im Anschluss auf dessen Struktur modelliert. Nach dem Modellieren der Struktur wird die neue Struktur an der Ausgangsstruktur ausgerichtet, damit die beiden Strukturen in der gleichen räumlichen Ausrichtung vorliegen und deren Elektrostatik miteinander verglichen werden kann.

Berechnen der Elektrostatik des Individuums Mit einer räumlich korrekt ausgerichteten Struktur des Individuums und der Referenzhülle wird nun die Elektrostatik an den Hüllenpunkten basierend auf dem gegebenen Individuum berechnet. Die ersten Schritte sind wie schon beschrieben das Hinzufügen der Radien und der Ladungen durch PDB2PQR und das anschließende Lösen der Poisson-Boltzmann-Gleichung durch APBS. Im Anschluss daran werden mit Hilfe der gegebenen Hülle die passenden Werte ausgewählt und eine zweite Hülle erstellt, die nun die Elektrostatikwerte

des Individuums enthält.

Vergleich zweier elektrostatischer Hüllen Der eigentliche Vergleich dieser Fitnessfunktion basiert auf den beiden Hüllen, deren Werte nun miteinander verglichen werden. Berechnet wird das Mittel der Differenz aller Wertpaare zweier Hüllen. Seien also A und B die zu vergleichenden Hüllen mit jeweils n Punkten a und b, so wird der Fitnesswert f berechnet mit:

$$f = \frac{1}{n} \sum_{i=1}^{n} |a_i - b_i| \tag{3.13}$$

Für den Vergleich der Optimierungsleistung der beiden Varianten der Fitnessfunktionen werden auch mit der Epitopsy-Version drei Optimierungen mit den zuvor gefundenen Parametern durchgeführt. Weil vor allem die DNA-Bindeleistung der Epitopsy-Variante mit der der Molekulargewichts-Variante verglichen werden soll, werden die Pareto-Fronten der Epitopsy-Optimierungen wie die der Optimierung basierend auf dem Molekulargewicht mit dem in Kapitel 2 entwickelten Programmablauf genauer untersucht.

3.6 Anpassen des bestehenden Verfahrens

Zum Ende dieses Kapitels soll nun die Optimierung des GAs mit geänderten Parametern und Operatoren überprüft werden, indem die Individuen aus der letzten Pareto-Front die in Absatz 2.5 und 2.6 beschriebenen Schritte durchlaufen. Einige Details werden dazu jedoch angepasst.

Um die Anzahl an GROMACS-Simulationen zu begrenzen, werden die Sequenzen der insgesamt sechs Pareto-Fronten mit ERIS bewertet und die jeweils besten fünf eines Laufes in einer MD simuliert. Im Anschluss findet eine manuelle Auswahl der drei stabilsten Strukturen der beiden Fitnessvarianten statt. Diese sechs Strukturen werden dann mit einer BrownDye- und AMBER-Simulation bewertet, um Rückschlüsse auf die Optimierungen des GAs aus diesem Kapitel schließen zu können.

3.7 Ergebnisse

3.7.1 Die optimalen GA-Parameter

Von den insgesamt 18 GA-Simulationen zur Parameteroptimierung wurden Konvergenzgrafiken erstellt und in Anhang C abgebildet. Jede Grafik zeigt zu einem Parametersatz den Verlauf der Fitnesswerte, genauer des maximalen, minimalen und

Parametersatz	Lauf 1	Lauf 2	Lauf 3	Mittel	Standardabweichung
(2000, 300, 005)	621	1477	1030	1042,67	428,14
(2000, 600, 005)	496	711	567	591,33	109,55
(2000, 300, 01)	680	660	628	656	26,23
(2000, 600, 01)	922	973	463	786	280,89
(2000, 300, 02)	652	543	720	638,33	89,29
(2000, 600, 02)	447	383	478	436	48,45

Tabelle 3.1: Aufstellung der Generationen, für die das LSSC in den Optimierungsläufen zur Parameteroptimierung eine Konvergenz angezeigt hat. Jedem Parametersatz aus Generationenzahl, Zahl der Individuen und Mutationsrate werden die drei Läufe sowie das daraus resultierende Mittel und die Standardabweichung zugeordnet.

mittleren Fitnesswertes einer gesamten Generation. In Rot wird der Sekundärstrukturwert, in Grün der Hydrophobizitäts- und in Blau der Molekulargewichtswert dargestellt (vgl. Abb. 2.10 aus Abs. 2.7.1). Zusätzlich wird die Generation durch eine schwarze Linie gekennzeichnet, für die das LSSC eine Konvergenz angibt. Für die Normierung der Grafiken auf den Bereich [0,1] wurde das Maximum der Fitnessfunktionen über alle 18 Simulationen bestimmt. Somit sind die Grafiken untereinander direkt vergleichbar.

Tabelle 3.1 zeigt die Generationen, für die LSSC eine Konvergenz bestimmt hat, sowie den Mittelwert und die Standardabweichung zum Mittelwert. Der Parametersatz wird dabei als Tupel angegeben, zum Beispiel `(2000, 600, 01)`. An erster Stelle steht dabei die Anzahl der Generationen, an zweiter Stelle die Anzahl der verwendeten Individuen und an letzter Stelle die Nachkommastellen der Mutationsrate.

Die Simulationen mit der niedrigen Mutationsrate `(2000, 300, 005)` und `(2000, 600, 005)` zeigen erwartungsgemäß eine langsame Konvergenz nach durchschnittlich 1043 respektive 591 Generationen. Letzterer Wert ist im Verhältnis zu den anderen Simulationen auffällig kurz, bei der Betrachtung der zugehörigen Graphen fällt jedoch auf, dass sich die Population nach dem vorhergesagten Konvergenzpunkt weiter verändern. Es ist offensichtlich noch keine Konvergenz eingetreten. Die Simulation mit 300 Individuen zeigt einen gleichmäßigen Optimierungsverlauf. Nach dem Konvergenzpunkt sind keine signifikanten Änderungen mehr zu beobachten. Bei 600 Individuen treten während der Simulation lange Phasen auf, in denen sich die Fitnesswerte nicht verändern, später jedoch noch eine Änderung auftritt. Dieses Verhalten kann offensichtlich nicht von dem LSSC vorhergesehen werden, ohne die Fensterbreite des Algorithmus drastisch zu erhöhen. Das hätte den Nachteil, dass eine gleichmäßige

Konvergenz erst sehr spät angezeigt wird. Die Streuung der festgestellten Konvergenz fällt ebenfalls hoch aus, verglichen mit den anderen vier Parametersätzen.

Bei genauerer Betrachtung der Läufe 1 und 2 des Satzes `(2000, 300, 005)` fällt auf, dass das Maximum der Sekundärstrukturfitness nicht erreicht wurde. Dies ist ein sehr starkes Kriterium was gegen den Parametersatz spricht, da die restlichen 16 Läufe dieses Maximum stets schon vor der 100. Generation erreicht haben.

Die Parametersätze `(2000, 300, 01)` und `(2000, 600, 01)` lassen eine schnellere Konvergenz beobachten als die Simulationen mit einer Mutationsrate von 0,005. Die Mittelwerte sind hier geringer (628 und 786 zu 1043 und 591) mit insgesamt geringerer Streuung. Dabei soll die Konvergenzfehlvorhersage des Parametersatzes `(2000, 600, 005)` nicht außer Acht gelassen werden.

Wieder sind in den Generationen nach angezeigter Konvergenz kleine Änderungen zu beobachten, wenn auch nicht so stark wie in den zwei vorherigen Parametersätzen. Das Zusammenspiel zwischen Mutationsrate und Generationenzahl hat sich bei dem Parametersatz `(2000, 600, 01)` verbessert. Er zeigt schnelle Konvergenz mit wenigen nachträglichen Änderungen. Diese nachträglichen Änderungen sind ein Indiz für zu wenig Individuen oder zu geringe Mutationsraten.

Bei der Betrachtung der Sätze `(2000, 300, 02)` und `(2000, 600, 02)` bestätigt sich diese Annahme. Wie die anderen Sätze mit 300 Individuen zeigt sich auch beim Satz `(2000, 300, 02)` eine relativ langsame Konvergenz. Daraus resultiert eine Anzeige einer Konvergenz vor der reellen Konvergenz des GAs durch eine zu geringe Fensterlänge des LSSC. Der Parametersatz `(2000, 600, 02)` zeigt das optimale Bild einer erfolgreichen Konvergenz unter den gegebenen Parametern und Operatoren. Nach einer schnellen, gleichmäßigen und durch das LSSC korrekt angezeigten Konvergenz finden kaum noch Änderungen in den Fitnesswerten statt. Das Mittel der Fitnesswerte bleibt gleich, es werden lediglich die Individuen mit den jeweils schlechtesten Fitnesswerten verbessert; einen relevanten Einfluss auf den Optimierungsausgang haben diese Änderungen nicht mehr.

Die von dem Parametersatz `(2000, 600, 02)` gezeigte Optimierungsleistung erlaubt eine Reduzierung der Generationenzahl auf 1000 und folglich eine Halbierung des Rechenaufwands im Gegensatz zu der Optimierung aus Kapitel 2.

3.7.2 Die optimalen genetischen Operatoren

Um einen passenden Satz von genetischen Operatoren zu finden (vgl. Abs. 3.4), wurden für die sechs verschiedenen Operatorsätze jeweils drei Optimierungen im GA gestartet. Die Grafiken zum Verlauf der Fitnesswerte sind abgebildet im Anhang D.

In den Grafiken der nicht elitären Simulationen ist jeweils der RMSF der drei Fitnesswerte vermerkt und die gemittelten Referenzpunkte durch eine schwarze, gestrichelte Linie gekennzeichnet. Auf Grund der Ergebnisse aus der vorherigen Optimierung der Parameter wurden alle Simulationen über 1000 Generationen mit 600 Individuen und einer Mutationsrate von 0,02 durchgeführt. Die unterschiedlichen Parametersätze werden wie in Absatz 3.4 mit BCP, CCP und PCP bezeichnet.

In Tabelle 3.2 sind die RMSF-Werte für die Läufe CCP und BCP jeweils mit und ohne eingeschränktes Aminosäurenalphabet dargestellt. Für den Parametersatz BCP wurde der RMSF nicht berechnet, da dieser keine Fluktuationen zeigt. Wie erwartet sind die RMSF-Werte im Mittel bei dem CCP-Operatorsatz um etwa 80% höher als die des PCP-Operatorensatzes. Interessanter ist jedoch das Verhältnis innerhalb eines Operatorensatzes mit und ohne eingeschränktes Aminosäurenalphabet. Hier bestätigt sich, dass die Einschränkung der Aminosäuren vor allem Einfluss auf die Größe der verwendeten Aminosäuren hat, folglich ihr Gewicht. Mit dem vollen Aminosäurenalphabet nimmt die Schwankung der Fitnesswerte für die Sekundärstruktur- und Hydrophobizitätsvorhersage ab. Grundlage dafür ist, dass der Algorithmus eine größere Auswahl an Aminosäuren hat und diese Eigenschaften so einfacher optimiert werden können. Im Gegenzug nimmt die Fluktuation der Fitnessfunktion auf Basis des Molekulargewichtes für beide Operatorensätze zu. Durch das Einschränken des Aminosäurenalphabetes wurden unter anderem sehr große Aminosäuren, wie zum Beispiel Tryptophan, ausgeschlossen. Diese führen bei der Molekulargewichtsfitness zu größeren Schwankungen und destabilisieren die Struktur gegebenenfalls. Gerade im Kern der Z-Domäne befinden sich überwiegend kleine Aminosäuren.

Ein direkter Vergleich der Konvergenzwerte zwischen den drei Operatorensätzen ist nicht sinnvoll, da die Parameter des LSSC wegen der unterschiedlich starken Fluktuationen neu angepasst werden mussten und diese Anpassung schon Grund für ein unterschiedliches Ergebnis sein kann. Hinzu kommt, dass kein Parameter für das LSSC gefunden werden konnte, der eine Konvergenz für den CCP-Operatorensatz anzeigt. So ist die Fluktuation während der Optimierung so stark, dass mit dem LSSC keine Konvergenz erkannt werden kann.

Abgesehen von der Fluktuation zeigen die Parametersätze CCP und PCP stabilere Simulationen als die BCP Simulation. Ein Indiz hierfür sind die gleichmäßigen Verläufe der Fitnesswerte über alle drei Läufe eines Parametersatzes. Während sich bei den elitären Simulationen die Werte für Hydrophobizität und Molekulargewicht bei jedem Lauf auf unterschiedlichem Niveau einpendeln, zeigen vor allem diese beiden Fitnessfunktionen bei den nichtelitären Sätzen innerhalb der drei Läufe eine Konvergenz im gleichen Wertebereich. Wie schon zuvor sind die Grafiken auf das Maximum

		Lauf 1	Lauf 2	Lauf 3	Mittel	Std.Abw.
CCP	SSP	$3{,}95 \cdot 10^{-2}$	$4{,}61 \cdot 10^{-2}$	$3{,}12 \cdot 10^{-2}$	$3{,}89 \cdot 10^{-2}$	$7{,}50 \cdot 10^{-3}$
	Hydro	$3{,}75 \cdot 10^{-5}$	$4{,}49 \cdot 10^{-5}$	$2{,}85 \cdot 10^{-5}$	$3{,}70 \cdot 10^{-5}$	$8{,}18 \cdot 10^{-6}$
	Mol	$2{,}31 \cdot 10^{-6}$	$2{,}88 \cdot 10^{-6}$	$1{,}77 \cdot 10^{-6}$	$2{,}32 \cdot 10^{-6}$	$5{,}54 \cdot 10^{-7}$
CCP AS	SSP	$3{,}13 \cdot 10^{-2}$	$4{,}24 \cdot 10^{-2}$	$4{,}05 \cdot 10^{-2}$	$3{,}80 \cdot 10^{-2}$	$5{,}96 \cdot 10^{-3}$
	Hydro	$3{,}10 \cdot 10^{-5}$	$3{,}94 \cdot 10^{-5}$	$3{,}30 \cdot 10^{-5}$	$3{,}45 \cdot 10^{-5}$	$4{,}38 \cdot 10^{-6}$
	Mol	$4{,}22 \cdot 10^{-6}$	$4{,}91 \cdot 10^{-6}$	$3{,}01 \cdot 10^{-6}$	$4{,}05 \cdot 10^{-6}$	$9{,}58 \cdot 10^{-7}$
PCP	SSP	$1{,}39 \cdot 10^{-2}$	$3{,}25 \cdot 10^{-2}$	$2{,}25 \cdot 10^{-2}$	$2{,}30 \cdot 10^{-2}$	$9{,}34 \cdot 10^{-3}$
	Hydro	$2{,}32 \cdot 10^{-5}$	$2{,}72 \cdot 10^{-5}$	$2{,}67 \cdot 10^{-5}$	$2{,}57 \cdot 10^{-5}$	$2{,}19 \cdot 10^{-6}$
	Mol	$1{,}95 \cdot 10^{-6}$	$8{,}28 \cdot 10^{-7}$	$1{,}28 \cdot 10^{-6}$	$1{,}36 \cdot 10^{-6}$	$5{,}66 \cdot 10^{-7}$
PCP AS	SSP	$1{,}57 \cdot 10^{-2}$	$8{,}98 \cdot 10^{-3}$	$1{,}60 \cdot 10^{-2}$	$1{,}35 \cdot 10^{-2}$	$3{,}96 \cdot 10^{-3}$
	Hydro	$1{,}89 \cdot 10^{-5}$	$2{,}48 \cdot 10^{-5}$	$2{,}16 \cdot 10^{-5}$	$2{,}18 \cdot 10^{-5}$	$2{,}95 \cdot 10^{-6}$
	Mol	$1{,}70 \cdot 10^{-6}$	$3{,}62 \cdot 10^{-6}$	$1{,}88 \cdot 10^{-6}$	$2{,}40 \cdot 10^{-6}$	$1{,}06 \cdot 10^{-6}$

Tabelle 3.2: RMSF der Läufe zur Operatorenoptimierung.

aller 18 Simulationen normiert und somit untereinander direkt vergleichbar.

Die Simulationen mit dem Operatorensatz BCP zeigen wie schon in Absatz 3.7.1 eine hohe Instabilität. Trotz der gleichen Parameter wie zuvor erreicht die Simulation mit eingeschränktem Aminosäurenalphabet nicht dieselbe Konvergenzgeschwindigkeit wie bei der Parameteroptimierung. Daraus wird offensichtlich, wie sehr die Optimierung mit elitären Operatoren zufallsbedingt variiert. Auch auf Grund sehr unterschiedlicher Simulationsziele ist dieser Operatorensatz nicht geeignet für die hier gestellte Optimierungsaufgabe.

Für die folgenden Untersuchungen ist wegen der geringeren Fluktuation, einem besseren absoluten Optimierungsergebnis und der Möglichkeit, die Konvergenz durch das LSSC Verfahren vorherzusagen, der PCP-Operatorensatz den anderen beiden Operatorensätzen vorzuziehen. Da die Fitnessfunktion auf Basis des Molekulargewichtes durch eine Vorhersage des elektrostatischen Potentials der Individuen ersetzt werden soll, (vgl. Abs. 3.5) muss weiterhin mit einem eingeschränkten Aminosäurenalphabet optimiert werden, um sterischen Problemen schon während der Optimierung entgegenzuwirken.

3.7.3 Epitopsy als Fitnessfunktion

Wie in Absatz 3.5 beschrieben wurden drei Optimierungen mit dem GA durchgeführt, bei denen als dritte Fitnessfunktion Epitopsy als Ersatz für die Molekulargewichtsvorhersage eingesetzt wird. Auf Grund der Ergebnisse aus den vorherigen Kapiteln wurde über 1000 Generationen mit 600 Individuen und einer Mutationsrate von 0,02 simuliert. Als genetische Operatoren wurden eine zufällige Auswahl der Eltern für die Rekombination und eine Roulette-Selektion zum Erstellen der Nachfolgegenerationen (PCP) genutzt.

Die Ergebnisse dieser drei Läufe sind in Abbildung 3.2 wie zuvor in Form eines Konvergenzgraphen abgebildet. Als erstes fällt das geänderte Verhalten der dritten Fitnessfunktion auf. Auf Grund der veränderten Methodik von Epitopsy gegenüber der Molekulargewichtsvorhersage, die sehr ähnlich zu der Hydrophobizitätsvorhersage ist, werden mehr Informationen über die spätere Struktur der Sequenz des einzelnen Individuums eingebracht. Daher zeigt diese Fitnessfunktion ein geänderten Fitnessverlauf. Die Konvergenzgeschwindigkeit ist jedoch vergleichbar mit der vorangegangener Simulationen. Durch die optimierten Parameter und Operatoren ist der Verlauf der Fitnesswerte gleichmäßig und über alle drei Läufe konsistent.

Im Vergleich zu den drei Optimierungsläufen mit gleichem Parameter- und Operatorensatz, jedoch mit der Molekulargewichtsvorhersage an Stelle von Epitopsy, zeigen diese Optimierungen andere RMSF-Werte. Der absolute RMSF der Epitopsy-Fitnessfunktion ist sehr klein; dies hat konzeptionelle Gründe. Vergleicht man jedoch die anderen beiden Fitnessfunktionen mit den Molekulargewichtsläufen, fällt auf, dass der RMSF der Sekundärstrukturvorhersage insgesamt gesunken, der der Hydrophobizitätsvorhersage jedoch stark gestiegen ist. Demnach hat entweder die Molekulargewichtsvorhersage die Hydrophobizitätsvorhersage stabilisiert oder sie wird nun von Epitopsy destabilisiert.

Die stärkeren Fluktuationen sind ebenso der Grund, warum für einen der Läufe keine Konvergenz mehr angezeigt wird, wie es schon bei der Operatorensuche für den Operator, der die gesamte Kindergeneration als Folgegeneration übernimmt, der Fall war.

Ebenfalls interessant ist die Zahl der Individuen auf der Pareto-Front am Ende eines Optimierungslaufes. Die bisherigen Läufe lieferten eine Anzahl von Individuen im Bereich von 57 bis 167. Mit 16 bis 27 Individuen zeigen die Optimierungläufe mit Epitopsy signifikant weniger Individuen auf der letzten Pareto-Front.

Insgesamt führt die Verwendung von Epitopsy zu einer Verachtfachung des Rechenaufwandes der gesamten GA-Optimierung. Die bisherigen Ergebnisse lassen auf leicht

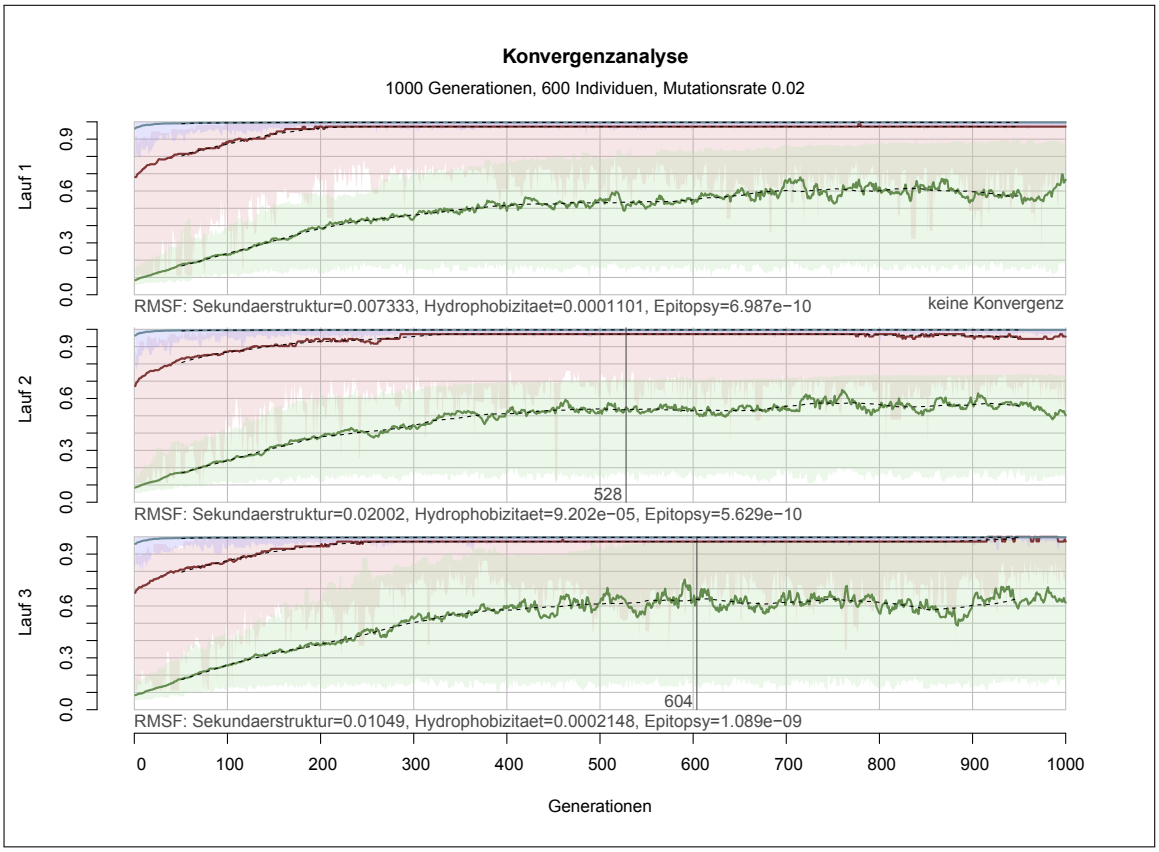


Abbildung 3.2: Konvergenzverhalten der GA-Optimierungen mit Epitopsy als zusätzlicher Fitnessfunktion an Hand der drei Fitnesswerte der Individuen auf der Pareto-Front einer jeden Generation. Die dargestellten Werte sind auf das jeweilige Maximum beider Simulationen normiert und auf den Bereich zwischen 0 und 1 skaliert. In Rot dargestellt ist der Sekundärstruktur-Wert, in Grün der Hydrophobizitäts- und in Blau der Epitopsy-Wert. Die farbige Linie gibt den mittleren Fitnesswert an, der farbige hellere Bereich markiert das Minimum und Maximum der Fitnesswerte.

instabilere Individuen schließen. So müssen die folgenden Analysen zeigen, ob der erhöhte Rechenaufwand durch eine höhere DNA-Bindefähigkeit gerechtfertigt wird.

3.7.4 Auswertung der GA-Optimierung

ERIS-Ergebnisse Um genauere Aussagen über die Tauglichkeit der Optimierungen dieses Kapitels treffen zu können, haben die Ergebnisse der letzten sechs GA-Optimierungen die in Absatz 2.5 und 2.6 beschriebenen Schritte durchlaufen. Als erstes wurden die Pareto-Fronten der letzten Generation jeder Optimierung ERIS übergeben. Anhand der Rangfolge basierend auf dem von ERIS berechneten $\Delta\Delta G$-Wert wurden die besten 5 Individuen eines jeden Optimierungslaufes gewählt. Eine

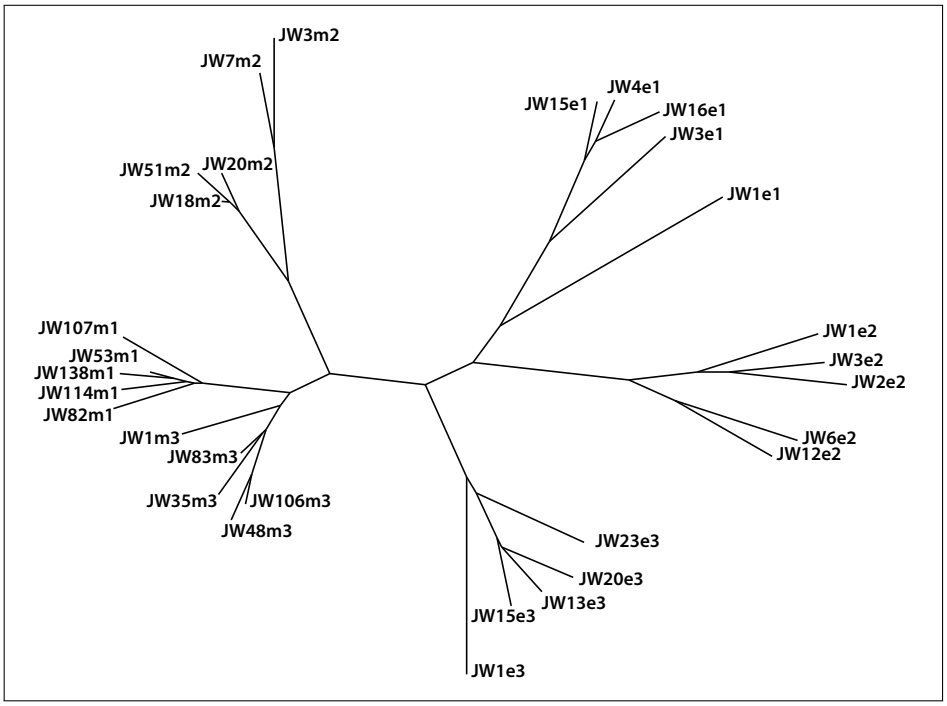


Abbildung 3.3: Phylogenetischer Baum basierend auf einem multiplen Sequenzalignment der 30 Individuen nach ERIS mittels ClustalW2 [50]. Die Identifikatoren entsprechen denen aus Anhang E. Der Baum wurde mit Phylodendron [31] visualisiert.

Tabelle mit den Sequenzen, den von ERIS berechneten Werten und den im Folgenden benutzten Identifikatoren dieser insgesamt 30 Individuen befindet sich in Anhang E.

Abbildung 3.3 zeigt einen wurzellosen, phylogenetischen Baum nach dem multiplen Sequenzalignment der 30 Sequenzen mit ClustalW2 [50]. Insgesamt zeigt der Baum, dass jeweils drei Läufe der beiden Varianten der Fitnessfunktionen ein eigenes Cluster bilden. Auch innerhalb dieser zwei Cluster bilden die drei einzelnen GA-Simulationen eigene Cluster aus. Die Cluster mit der Molekulardynamik-Fitnessfunktion sind zueinander jedoch ähnlicher als die Läufe mit der Epitopsy-Fitnessfunktion. Vor allem die m1 und m3 Cluster liegen sehr nahe beieinander und weisen selbst kürzere Distanzen auf.

Die Molekulardynamik-Läufe finden also insgesamt Ergebnisse, die sich auch über mehrere Läufe ähnlicher zueinander sind, als die der Epitopsy-Läufe. Die Konvergenz auf ein globales Minimum hin ist somit insgesamt weiter fortgeschritten. Es werden gegenüber der Epitopsy-Fitnessfunktion konstantere und besser reproduzierbare Ergebnisse erzeugt.

GROMACS-Ergebnisse Die auf Basis der ERIS-Rangfolge gewählten 30 Strukturen wurden 20 ns mit den in Absatz 2.5.2 beschriebenen Parametern simuliert. Die

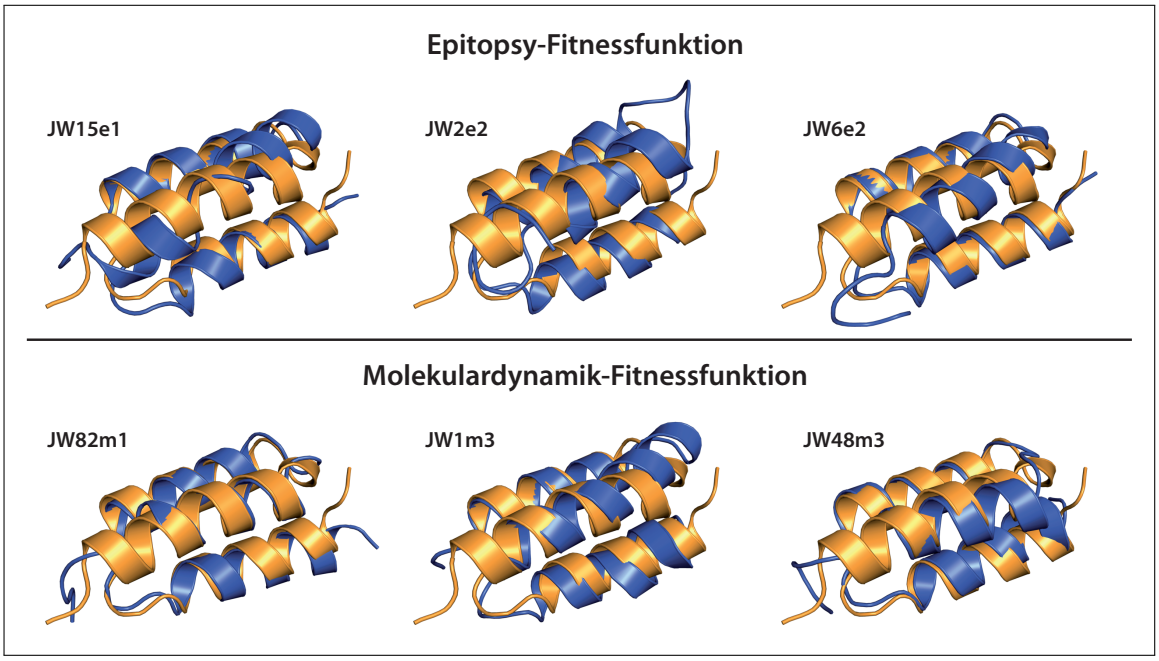


Abbildung 3.4: Strukturvergleich des letzten Simulationsschrittes der sechs ausgewählten Strukturen (blau) ausgerichtet an der Zielstruktur der Z-Domäne (PDB-ID: `1LP1`) (orange). Zur zukünftigen Identifikation wurden die Strukturen mit Bezeichnern versehen (links oben an jeder Struktur). Die dritte Helix der Strukturen befindet sich in den Abbildungen im Vordergrund, der N-Terminus somit oben rechts.

Simulationen der Strukturen JW4e1 und JW15e3 wurden nicht korrekt beendet. Bei der Simulation wurde der Fehler gemeldet, dass sich zwischen zwei Schritten des Gebietszerlegungsalgorithmus eine geladene Gruppe zu viel bewegt hat. Der Grund für diesen Fehler wurde nicht genauer untersucht. Die beiden Simulationen, in denen er aufgetreten ist, wurden verworfen.

Die verbleibenden 28 Simulationstrajektorien wurden manuell untersucht und die drei Individuen der zwei Fitnessfunktionsvarianten mit der jeweils geringsten Dislokation der Helices wurden für die weiteren Untersuchungen ausgewählt. Abbildung 3.4 zeigt die ausgewählten Strukturen ausgerichtet an der Zielstruktur, der Z-Domäne.

Schon bei den jeweils drei stabilsten Strukturen der Simulationen fällt auf, dass die Strukturen aus der Optimierung mit der Molekulargewichts-Fitnessfunktion die Struktur der Z-Domäne besser halten als die Strukturen mit Epitopsy als Fitnessfunktion. Dies gilt auch bei Betrachtung aller simulierter Strukturen. Insgesamt zeigen die Strukturen aus der Molekulargewichts-Optimierung weniger strukturelle Abweichungen zur Z-Domäne.

Die drei Molekulargewichts-Strukturen sind mit der der Z-Domäne beinahe identisch. Die Abstände der Helices werden sehr gut über die Simulationsdauer konser-

viert. In JW81m1 bleibt zusätzlich zu den Helices die Struktur der flexiblen Schleifen zwischen den Helices erhalten. Jedoch entfaltet sich die dritte Helix leicht in der letzten Umdrehung. Eine ähnliche Entfaltung zeigt JW48m3, kann zusätzlich jedoch die Struktur der flexiblen Schleife zwischen Helix 2 und 3 nicht halten. JW1m3 zeigt einen leichten Drift in der dritten Helix am Übergang zur zweiten Helix.

Für die Strukturen der Epitopsy-Fitnessfunktion sind im Gegensatz zu denen der Molekulargewichts-Fitness mehr Strukturänderungen zu beobachten. JW1e1 zeigt einen Bruch der helicalen Struktur in der dritten Helix in Folge einer Verschiebung der Enden in unterschiedliche Richtungen. JW2e2 zeigt Verschiebungen der Fc-bindenen Helices und wie schon zuvor beobachtet ein Entwinden der dritten Helix. JW6e2 kann die Abstände der Helices zueinander konservieren. Hier zeigt sich lediglich eine Rotation der dritten Helix.

Die drei Strukturen der Molekulargewichts-Optimierung zeigen eine sehr hohe Stabilität über die Simulation, die der von JW19 sogar überlegen scheint. Die Struktur der Helices zueinander wird perfekt konserviert, teilweise werden sogar die variablen Schleifen an ihrem Platz gehalten. An diesen Grad der Stabilität können die Strukturen der Epitopsy-Optimierung nicht anknüpfen und befinden sich damit etwa auf Höhe von JW70. Insgesamt übertreffen die Ergebnisse die Ergebnisse der ersten GA-Optimierung aus Kapitel 2.

BrownDye-Ergebnisse Tabelle 3.3 zeigt eine Auflistung der mit den sechs ausgewählten Strukturen durchgeführten BrownDye-Simulationen. Als Referenz sind einige Werte aus der Tabelle 2.2 aus Absatz 2.7.4 nochmals aufgeführt. Die sechs BrownDye-Simulationen wurden wie in Absatz 2.6.1 beschrieben durchgeführt.

Die Epitopsy-Strukturen erreichen relative k_{on}-Werte von 25-85%, die Molekulargewichtsstrukturen Werte von 58-134%. Erstere liegen damit alle zwischen der besten (JW70) und zweitbesten (JW56) Struktur aus der ersten Optimierung und über der Startstruktur. Somit war in allen Fällen die Optimierung lohnend im Hinblick auf die DNA-Bindung, was bei der ersten Optimierung nur bei JW70 der Fall war. Der beste relative k_{on} wird hier nicht von der Struktur mit der größten Nettoladung erreicht. Das bestätigt die Annahme aus Kapitel 2, dass die Nettoladung zwar problembedingt für eine gute oder schlechte Assoziationsrate ausschlaggebend ist, jedoch nicht das alleinige Kriterium dafür darstellt. Mit 85% relativem k_{on} bewegt sich JW15e1 im Bereich von JW70 und der Referenzstruktur, also der MyoD-DNA-Bindehelix.

Die Strukturen der Molekulargewichts-Optimierung erreichen höhere Werte als die der Epitopsy-Optimierung. Mit 134% relativem k_{on} findet sich im Fall von JW82m1 eine Struktur, die eine höhere Assoziationsrate aufweist als die Referenzstruktur. Auch

Struktur	$k_{on}(M^{-1}s^{-1})$	Nettoladung	k_{on}^{rel}(Wildtyp)
DNA-Bindehelix (Wildtyp)	$4{,}59 \cdot 10^8$	+5	1,000
Z-Domäne (Negativkontrolle)	0	-2	0,000
Startstruktur	$1{,}04 \cdot 10^8$	+5	0,227
JW70	$4{,}21 \cdot 10^8$	+5	0,917
JW15e1	$3{,}94 \cdot 10^8$	+4	0,854
JW2e2	$1{,}15 \cdot 10^8$	+6	0,251
JW6e2	$1{,}77 \cdot 10^8$	+5	0,386
JW82m1	$6{,}16 \cdot 10^8$	+5	1,342
JW1m3	$4{,}15 \cdot 10^8$	+5	0,904
JW48m3	$2{,}65 \cdot 10^8$	+5	0,577

Tabelle 3.3: BrownDye-Ergebnisse der sechs besten Individuen aus der GROMACS-Simulation.

JW1m3 weist mit einem k_{on} von $4{,}15 \cdot 10^8 M^{-1}s^{-1}$ eine Assoziationsrate auf, die mit der der Referenzstruktur gleichzusetzen ist. Alle drei Strukturen weisen eine Nettoladung von +5 auf, wie es schon bei JW70 und der Startstruktur der Fall.

AMBER-Ergebnisse Alle sechs Strukturen, die auch die BrownDye-Simulation durchlaufen haben, wurden auch in AMBER simuliert, um die Komplexe auf ihre Dissoziation zu überprüfen. Verfahren wurde nach der in Absatz 2.6.2 beschriebenen Methode. Die Simulationen ergaben nach 10 ns Simulationszeit das in Abbildung 3.5 gezeigte Bild.

Die drei Strukturen aus der Optimierung mit Epitopsy als dritter Fitnessfunktion verbleiben über die Simulationszeit an der Stelle, an der sie an die DNA modelliert wurden. Wie schon bei JW70 werden leichte Verschiebungen der ersten und zweiten Helix zur dritten Helix hin beobachtet. Entfaltungen wie bei der Startsequenz oder teilweises Lösen der Struktur von der DNA ist bei diesen drei Strukturen nicht zu beobachten (vgl. Abs. 2.7.5).

Die drei Strukturen mit dem Molekulargewicht als Teil der Fitnessfunktionen zeigen keine starke Bindung zur DNA. JW1m3 verbleibt über die Simulationsdauer zwar an der DNA, bei Betrachtung der Simulationstrajektorie ist jedoch in der ersten Hälfte der Simulation der Trend zu erkennen, dass sich das Protein von der DNA löst. Im weiteren Verlauf der Simulation zeigt sich zwar eine stabile Bindung zur DNA, auf Grund der Entfernung zur eigentlichen Bindestelle ist es jedoch fraglich, ob die Bindung über längere Zeit aufrecht gehalten werden kann. Die anderen zwei

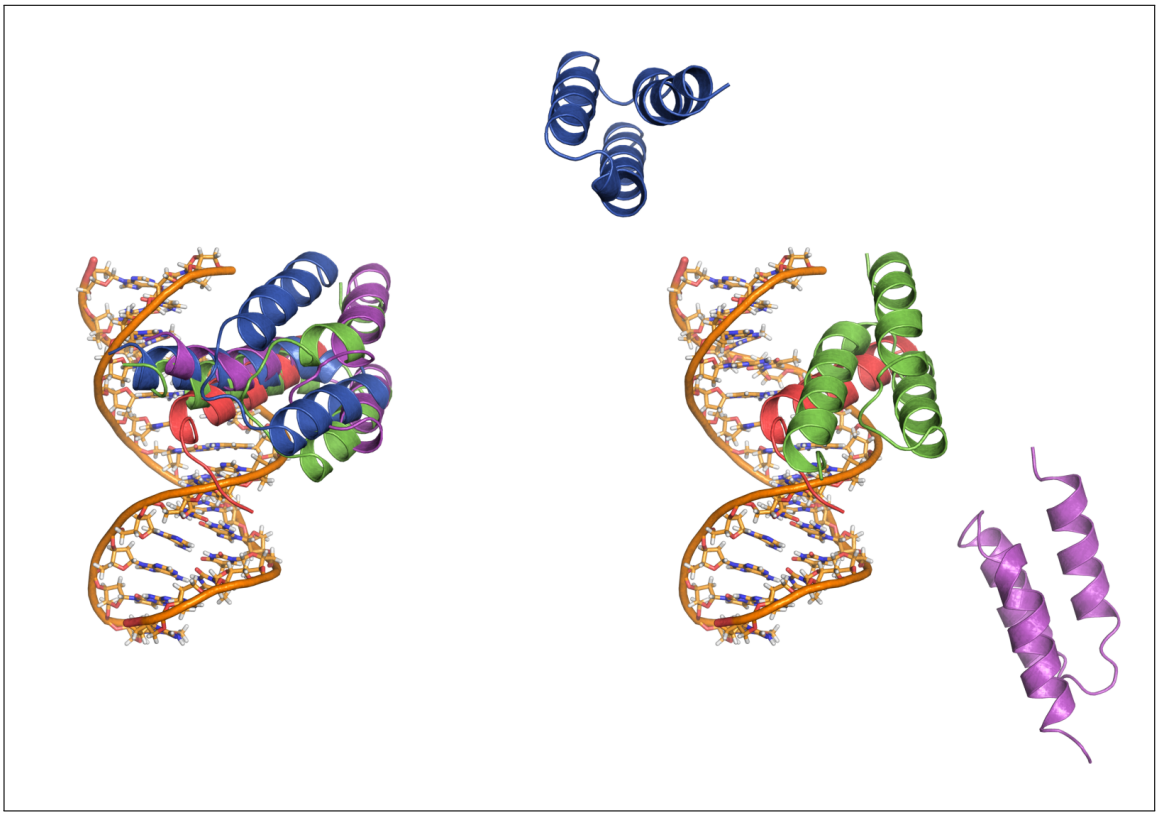


Abbildung 3.5: Ergebnis der sechs AMBER-Simulationen. Ausgerichtet an der DNA (orange) wurden die Strukturen der MyoD-DNA-Bindehelix (rot), sowie auf der linken Seite: JW15e1 (blau), JW2e2 (grün) und JW6e2 (violett) dargestellt. Auf der rechten Seite: JW82m1 (blau), JW1m3 (grün) und JW48m3 (violett).

Strukturen haben während der Simulation komplett von der DNA gelöst und sind von ihr weg diffundiert.

3.8 Diskussion

In diesem Kapitel wurden verschiedene Schritte unternommen, um die Leistung des GAs zu verbessern und die Konsistenz der gelieferten Ergebnisse zu erhöhen. Dazu wurden in einem ersten Schritt optimale Parameter für den GA gesucht. Untersucht wurden verschiedene Werte für die Generationenzahl, die Anzahl an Individuen und die Mutationsrate. In einem weiteren Schritt wurden verschiedene Varianten für die Selektion innerhalb des GAs sowie die Auswirkung des eingeschränkten Aminosäurenalphabets untersicht.

Dieser zweigeteilte Findungsprozess stellt natürlich nicht das Optimum dar, um die beste Konstellation der Parameter zu finden. So könnte nun mit geänderter Selektion eine andere Anzahl von Individuen und Generationen bessere Ergebnisse liefern.

Auch kann die Anzahl an Generationen, Individuen und die Mutationsrate in feineren Schritten untersucht werden. Unter dieser Argumentation müsste der Kreis jedoch immer weiter geführt werden. Eine andere Möglichkeit, um für die fünf Eigenschaften die besten Ergebnisse zu erhalten, wäre eine parallele Untersuchung. Jedoch kämen wir mit den in diesem Kapitel genutzten Werten auf 108 Simulationen. Mit etwa einem Jahr an reiner Rechenzeit auf 30 Computersystemen ist dies keine praktikable Lösung und auch ein immer abwechselndes Optimieren der Parameter kann auf Grund der Rechendauer nicht durchgeführt werden. Die Parameter sind folglich wohl nicht die besten, die für das Problem theoretisch existieren. Die Analysen der GA-Optimierungen zeigen jedoch, dass die durchgeführten GA-Optimierungen denen aus Kapitel 2 überlegen sind.

Zur Steigerung der Konsistenz über mehrere Läufe hat vor allem die geänderte, nicht mehr elitäre Fitnessfunktion beigetragen. Die konzeptionell dadurch bedingten Fluktuationen in den Fitnesswerten der Individuen machen das Bestimmen einer Konvergenz der Optimierung natürlich schwieriger. Durch Einstellen der LSSC-Parameter auf diesen neuen Umstand wurde jedoch eine zuverlässige, mathematisch fundierte Konvergenzvorhersage erreicht, die in Zukunft auch schon während der Optimierung für einen Stopp der Rechnung sorgen kann. Um jedoch falsche Einstellungen zu erkennen, wurde in dieser Arbeit mit einer festen Generationenzahl gerechnet.

Der verwendete Operatoren-Satz ist eher unüblich. Gängige Ansätze wie zum Beispiel der NSGA-II [15] verwenden elitäre Ansätze und erreichen so sehr schnelle Konvergenz und mit Hilfe ein paar weiterer Tricks verlässliche Ergebnisse. Es hat sich jedoch gezeigt, dass sich ein elitärer Ansatz bei diesem Problem oft in suboptimale Minima festsetzt. Grund hierfür ist die Komplexität des Problems, was zu einer sehr rauhen Fehleroberfläche führt. Hier haben Ansätze, die auch schlechte Individuen in neue Generationen übernehmen, Vorteile, da dadurch eine jede Generation einen größeren Bereich auf der Fehleroberfläche abdeckt und lokale Minima übersprungen werden können.

Auffällig ist nach Einführen von Epitopsy als dritte Fitnessfunktion das Sinken der Anzahl an Individuen auf der Pareto-Front. Hierfür sind wohl gleich mehrere Effekte verantwortlich. Insgesamt wird eine im Verhältnis gute Pareto-Front von weniger, dafür aber sehr guten Individuen aufgespannt. Es macht den Anschein, als werden weniger Kompromisslösungen für die Pareto-Front gefunden. So entstehen per Zufall von Epitopsy besonders gut bewertete Individuen im Verhältniss zu ähnlichen Sequenzen.

Eine Eigenschaft von Epitopsy spricht für diese These. Das auf der gegebenen Hülle berechnete Potential ist stark von der Positionierung der Seitenketten abhängig.

Diese wird in jeder neuen Generation vor der Berechnung der Fitness durch das Modellieren der Sequenz auf die Struktur der Z-Domäne mittels Modeller festgelegt. Da es sich aber um einen teilweise stochastischen Prozess handelt und die Seitenketten selbst in der Natur größeren Schwankungen ausgesetzt sind, variieren diese auch in den Modellen von Modeller. Es ist durch diesen Effekt möglich, dass der Epitopsy-Wert von zwei gleichen Sequenzen mit unterschiedlich gepackten Seitenketten in deren Strukturen eine größere Differenz hat als der Epitopsy-Wert von zwei Strukturen, deren Unterschied mutationsbedingt ist.

Der Einfluss dieser Differenzen in den Epitopsy-Werten kann mathematisch leicht durch Mitteln mehrere Vorhersagen von Modellen mit unterschiedlich gepackten Seitenketten minimiert werden. Epitopsy sorgt jedoch mit Anteilig fast 90% der Rechenzeit für eine Verachtfachung der Rechendauer eines Optimierungslaufes auf etwa zwei Wochen. Mehrmaliges Ausführen von Epitopsy für jedes Individuum – etwa 10 Durchläufe, würde für einen nicht mehr praktikablen Rechenaufwand sorgen und ist darum zum jetzigen Zeitpunkt nicht durchführbar. Abgesehen von eiener kleineren Pareto-Front ist es jedoch wahrscheinlich, dass der Effekt der Abweichungen vom theoretischen Mittel einer jeden Epitopsy-Vorhersage durch die große Anzahl an Individuen in der GA-Optimierung geschmälert wird.

Die ERIS-Ergebnisse geben wenig Anlass zur Diskussion. Insgesamt fallen die absoluten $\Delta\Delta G$-Werte gering aus im Vergleich zu den ERIS-Werten aus Kapitel 2. Jedoch wurde dort schon bemerkt, dass die absoluten Werte kaum Aussagekraft über die in der GROMACS-Simulation gezeigte Stabilität besitzen. Nähern sich die Sequenzen mehrerer Läufe jedoch phylogenetisch an, werden auch die ERIS-Werte zueinander mehr und mehr vergleichbar. Wie stark sich dieser Effekt auswirkt, ist schwer zu beurteilen. ERIS gibt jedoch einen Hinweis darauf, dass sich die Optimierungsläufe zueinander mehr ähneln, als dies noch in Kapitel 2 der Fall war.

Bei den GROMACS-Simulationen haben die Strukturen der Optimierungen, die das Molekulargewicht beinhalten, höhere Stabilität gezeigt als die Strukturen der Epitopsy-Optimierung. Verglichen mit den Strukturen aus Kapitel 2 erzeugen beide Fitnessfunktionstypen Individuen mit einer hohen strukturellen Stabilität. Als Grund für die geringere Stabilität der Epitopsy-Gruppe ist offensichtlich Epitopsy verantwortlich. Bei genauerer Betrachtung der Strukturen fällt auf, dass durch Verlust der Molekulargewichtsvorhersage die Größe der Aminosäuren offensichtlich bei der Optimierung keine Rolle mehr spielt. Das verkleinerte Aminosäurenalphabet kann die Auswahl großer Aminosäuren für den hydrophoben Kern nicht ausreichend einschränken. So wird in der Optimierung mit Epitopsy auffallend oft Phenylalanin im Kern der Struktur platziert. Die hohe Hydrophobizität macht es der Optimierung einfach,

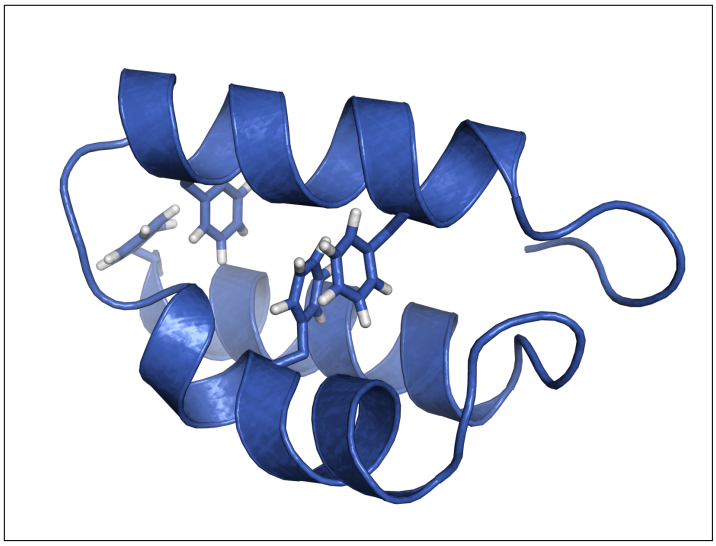


Abbildung 3.6: Struktur von JW13e3 nach der GROMACS-Simulation. Eingezeichnet sind zusätzlich die vier im Kern befindlichen Phenylalanine. Helix 3 befindet sich im Bild oben.

einen hohen Hydrophobizitätswert zu erreichen. Wie Abbildung 3.6 beispielhaft zeigt, wird so aus sterischen Gründen verhindert, dass sich die Sequenz zu der kompakten Struktur der Z-Domäne falten kann. Diese hingegen besitzt drei Phenylalanine, von denen jedoch nur eins in Kernnähe auftritt.

Insgesamt treten auf Grund dieses Effektes vermehrt Anhebungen der dritten Helix von den Helices 1 und 2 auf. Funktionell sollte dies nur wenig Einfluss nehmen, da die Bindestellen dadurch nicht beeinträchtigt werden. Bei der DNA-Bindung könnte sich dies sogar positiv auswirken, ist so mehr Platz um die dritte Helix für die DNA, in der die Helix komplett eingebettet wird. Es wird jedoch nicht das Ziel erreicht, eine Struktur möglichst übereinstimmend mit der Z-Domäne zu erzielen.

Auch die k_{on}-Werte der BrownDye-Simulationen haben sich gegenüber denen in Kapitel 2 verbessert. Jedoch wird hier der gegenteilige Effekt beobachtet, der von der Zusammensetzung der Fitnessfunktionen zu erwarten war. Die Epitopsy-Strukturen erreichen im Vergleich zu den Molekulargewicht-Strukturen schlechtere Assoziationswerte. Zu erwarten war der gegenteilige Fall. So wurde doch mittels Epitopsy die Elektrostatik der Proteine optimiert, die die Berechnungsgrundlage für BrownDye sind.

Betrachtet man jedoch den Versuchsaufbau von BrownDye genauer, gibt es Einschränkungen bezüglich der Aussagekraft. Betreffend der Elektrostatik spielen während einer BrownDye-Simulation vor allem die weitreichende, schwache Elektrostatik eine Rolle. Diese wird durch Epitopsy nicht direkt optimiert, arbeitet Epitopsy doch auf einer Hülle, die 6 Å von der SAS des Proteins entfernt ist. Zweitens bestimmt

BrownDye vor der Simulation Paare von Atomen, die einen Kontakt ausüben. Für diesen Schritt sind die DNA und die Proteinstruktur zueinander möglichst genau in der zu untersuchenden Bindestelle zu platzieren. Offensichtlich spielen auch hier die Seitenketten der dritten DNA-Bindehelix der jeweiligen Struktur des Fusionsproteins eine wichtige Rolle. Sind diese optimal zur DNA ausgerichtet, ist eine höhere Assoziationsrate zu erwarten. Kleine Schwankungen der k_{on}-Werte können somit schon alleine der Anordnung der Seitenketten geschuldet sein. Unterschiede, die etwa das Fünffache oder mehr betragen, lassen sich damit jedoch nicht mehr erklären. Gründe hierfür sind folglich beim Protein zu suchen.

In Kapitel 2 traten solch feine Unterschiede wie in diesem Kapitel nicht auf. Dort war das beste Individuum den anderen beiden um den Faktor 10 beziehungsweise 30 überlegen. Auf Grund der großen Streuung der BrownDye-Ergebnisse aus diesem Kapitel lassen sich jedoch kaum Aussagen treffen. Insgesamt scheinen jedoch die Sequenzen aus der Molekulargewichts-Optimierung höhere Assoziationsraten aufzuweisen.

Die AMBER-Simulationen liefern im Gegensatz zu den BrownDye-Simulationen eine Aussage über die Dissoziation. Die Strukturen der Molekulargewichts-Optimierung zeigen alle eine kaum vorhandene DNA-Bindeleistung. Die Epitopsy-Strukturen binden hingegen stabil an die DNA, wenn auch mit leichten strukturellen Deformationen zwischen der dritten Helix und den Helices 1 und 2. Schon methodisch ist diesen Simulationen eine höhere Aussagekraft zuzuordnen, als den BrownDye-Simulationen, können sich hier zum Beispiel schlecht positionierte Seitenketten während der Simulation neu Ausrichten und ihre unter den gegebenen Simulationsparametern optimale Position einnehmen.

Insgesam lässt sich nun also schlussfolgern, dass JW15e1 aus der Optimierung mit Epitopsy als dritter Fitnessfunktion für das hier gestelle Problem die vielversprechensten Eigenschaften besitzt: Eine insgesamt gute Stabilität in den GROMACS-Simulationen, eine mit 85% zum Wildtyp vergleichbare Assoziationsrate und sehr gute DNA-Bindeeigenschaften in der AMBER-Simulation.

Generell lässt sich festhalten, dass es keine eindeutig bessere Lösung gibt, was die Auswahl der dritten Fitnessfunktion angeht. Wird als Ziel der Optimierung vor allem die Stabilität des resultierenden Proteins festgelegt, so ist zu der Molekulargewichtsvorhersage als dritter Fitnessfunktion zu raten. Spielt die Elektrostatik im resultierenden Protein eine sehr wichtige Rolle, so ist Epitopsy als Fitnessfunktion einzusetzen. Diese Entscheidung hängt vom gestellten Problem ab und muss somit schon vor der Optimierung im GA überdacht werden. Insgesamt hat sich in diesem Kapitel jedoch gezeigt, dass der GA mit den hier entwickelten Optimierungen konstant

bessere Ergebnisse liefern kann, als es im Kapitel 2 der Fall war. Die durchgeführten Optimierungen sind somit förderlich für die Optimierungskraft des GAs.

4 Zusammenfassung und Ausblick

«Stay hungry. Stay foolish»

Stewart Brand

4.1 Zusammenfassung

In dieser Arbeit wurden erfolgreich zwei funktionale Regionen bestehender Proteine zu einem neuen bifunktionalen Protein vereint. Das Ersetzen eines Sequenzstückes einer Proteinstruktur durch die Sequenz, die eine neue Funktionalität bereitstellen soll, hat zu einer erhöhten Instabilität des Proteinkomplexes geführt. Folglich war der Erhalt der gewünschten Funktionen der beiden Bindestellen im Protein nicht mehr sichergestellt.

Eigenschaften wie die Sekundärstruktur, die Hydrophobizität und das Molekulargewicht der Aminosäuren sowie die Elektrostatik des Proteins wurden durch einen Genetischen Algorithmus dahingehend verändert, dass diese mit dem Referenzprotein, der Z-Domäne, so weit wie möglich übereinstimmen. Dadurch wurde vor allem die Stabilität der Aminosäurensequenz unter Erhaltung der neu eingebrachten Bindeeigenschaften erhöht.

Die Ergebnisse des GAs wurden unter Verwendung von vier unabhängigen Methoden verfeinert und getestet. So wurde aus der großen Menge an Individuen die mit den besten Stabilitäts- und Bindeeigenschaften identifiziert. Neben einer Stabilitätsprü-

fung durch GROMACS und ERIS wurde durch BrownDye und AMBER die Funktion der neu in die Z-Domäne eingebrachte DNA-Binderegion überprüft. Der Vergleich zur nicht optimierten Sequenz hat den Nutzen der Optimierung aufgezeigt. Verglichen mit dem Wildtyp der DNA-Binderegion zeigten die optimierten Strukturen vergleichbare Bindeeigenschaften.

Im Verlauf der Arbeit wurde an den Parametern des GAs, mit denen die ersten erfolgreichen Optimierungen durchgeführt wurden, eine Vielzahl von Veränderungen vorgenommen. Die Leistung des GAs wurde vor allem in Hinblick auf die Reproduzierbarkeit der Vorhersage verbessert. So liegen die resultierenden Pareto-Fronten mehrerer Simulationen auf der Fehleroberfläche enger zusammen als zuvor.

Auf Grund von Erkenntnissen aus den ersten Ergebnissen wurde der Nutzen einer weiteren Fitnessfunktion basierend auf der elektrostatischen Hülle der jeweiligen Proteinstruktur überprüft. Die Bindestärke der neu eingebrachten DNA-Bindestelle wurde durch diese Änderung gegenüber der Vergleichsgruppe deutlich erhöht. Durch Verlust der molekulargewichtsbasierten Fitnessfunktion kam es gleichzeitig zu einer Reduktion der Stabilität des gesamten Komplexes. Verglichen mit den ersten Ergebnissen wird dieser Effekt jedoch durch die Optimierung der Parameter ausgeglichen.

Die hohe Komplexität der Vorhersagemethode für die Elektrostatik erhöht den Rechenaufwand der gesamten Optimierung um etwa den Faktor acht. Somit ist diese Kombination von Fitnessfunktionen nur sinnvoll, wenn die Elektrostatik des Zielproteins notwendigerweise mit optimiert werden soll. Bei einer vorangigen Maximierung der Stabilität des Komplexes ist als dritte Fitnessfunktion die Molekulargewichtsvorhersage vorzuziehen.

Die Verwendung von leichtgewichtigen Fitnessfunktionen ermöglicht, das Potential des Genetischen Algorithmus durch eine vergleichbar große Anzahl an Individuen und Generationen gut auszunutzen. Die Vorhersagekraft des Algorithmus bleibt trotz geringer Rechenzeit durch die gelungene Auswahl an Fitnessfunktionen hoch. Die steigende Komplexität der nachfolgenden Methoden kann durch wiederholtes Aussortieren von schlechten Individuen kompensiert werden. Aus rund 100 potentiellen Individuen nach der GA-Optimierung wird so schrittweise eine kleine Anzahl der vielversprechendsten Sequenzen für die gestellten Anforderungen gefunden.

4.2 Ausblick

Die nun wohl wichtigste Frage nach vollendetem Optimierungs- und Auswahlprozess ist: Wenn auch alle Simulationen vielversprechende Ergebnisse liefern, besitzt das Protein unter Laborbedingungen genau die vorhergesagten Eigenschaften? Somit ist

der interessanteste Punkt für einen Ausblick, die gefundenen Sequenzen für JW82m1 und JW15e1 im Labor zu testen. Aufschlussreich sind hier Experimente, die Informationen über die dreidimensionale Stabilität des Protein geben, wie zum Beispiel eine CD-Spektroskopie oder das Erstellen einer NMR-Struktur. Vor allem die Frage, ob das Protein nach seiner Synthetisierung, ob nun *in vitro* oder *in vivo*, in die gewünschte Struktur faltet, kann dadurch geklärt werden. Ferner können Experimente zum Nachweis der Bindeleistungen an DNA und Fc weitere Indizien für den Erfolg des Optimierungsprozesses liefern.

Neben den labortechnischen Untersuchungen ist vor allem interessant, ob eine Kombination aller vier hier vorgestellten Fitnessfunktionen die Leistung des Algorithmus erhöht oder durch einen zu großen Suchraum verschlechtert.

Die dargestellte Methode zum Design von Proteinen besitzt großes Potential. Vorausgesetzt, die Optimierung skaliert gut auf größere Proteine mit einer längeren Aminosäurensequenz. So kann basierend auf einer vorgegebenen Struktur ein Protein um Funktionen erweitert werden. Durch Einfügen von strukturell verwandten Sequenzen in ein gegebenes Protein können beinahe beliebige Funktionen miteinander kombiniert werden. Die funktionalen Gruppen von Proteinen werden so zu Bausteinen eines Baukastensystems.

Durch die in den nächsten Jahren zu erwartende Steigerung der Rechenleistung von Computersystemen ergeben sich weitere Möglichkeiten zur Optimierung, die im Moment wegen des hohen Rechenaufwands unattraktiv wirken. Der Effekt einer größeren Individuenzahl pro Generation oder der Verwendung der UniRef90-Datenbank für die Sekundärstrukturvorhersage kann in diesem Szenario ausgiebig untersucht werden.

Ebenfalls im realistischen Zeitrahmen durchführbar wären dann Untersuchungen zur Konsistenz der Ergebnisse von Epitopsy in Bezug auf die Positionierung der Seitenketten der Proteinstruktur. So könnte Epitopsy wie bereits vorgeschlagen mehrmals ausgeführt werden. Die Ergebnisse von Modellen einer Sequenz mit unterschiedlicher Seitenkettenposition könnten dann gemittelt werden und zu stabileren Werten führen.

Die Molekulardynamiksimulationen würden ebenfalls von einer gesteigerten Rechenleistung profitieren. Besonders die Anzahl der Individuen, die durch die Simulationen untersucht werden, kann erhöht werden. Ebenso kann die Länge der Simulationen um einen gewissen Grad erhöht werden. Vor allem aber können mehrere Simulationen einer Struktur mit unterschiedlichem Startmodell durchgeführt werden. Die Aussagekraft der Simulationen würde durch diese Maßnahmen weiter um ein Vielfaches erhöht werden.

Es gibt also noch einige Stellen in diesem Projekt, an denen schon heute, an ma-

chen aber auch erst in der Zukunft, geforscht werden kann. Durch einen flexiblen Aufbau wurde der Grundstein für die Weiterentwicklung des Projektes gelegt, dessen Aussichten, und das verdeutlichen die bisherigen Ergebnisse, sehr vielversprechend sind.

Anhang

A Werte der Hydrophobizitäts- und Molekulargewichtsfitness

Auflistung der für die Hydrophobizitäts- und Molekulargewichtsvorhersage aus Absatz 2.4 verwendeten Werte.

Aminosäure	Hydrophobizitätswert	Molekulargewichtswert
Alanin	1,8	15,03
Cystein	2,5	47,10
Asparaginsäure	-3,5	59,04
Glutaminsäure	-3,5	73,07
Phenylalanin	2,8	91,13
Glycin	-0,4	1,01
Histidin	-3,2	81,1
Isoleucin	4,5	57,11
Lysin	-3,9	72,13
Leucin	3,8	57,11
Methionin	1,9	75,15
Asparagin	-3,5	58,06
Prolin	-1,6	42,08
Glutamin	-3,5	72,09
Glutaminsäure	-4,5	100,14
Serin	-0,8	31,03
Threonin	-0,7	45,06
Valin	4,2	43,09
Tryptophan	-0,9	130,16
Tyrosin	-1,3	107,13

B ERIS-Rangfolge der 1000 und 2000 Generationen GA-Simulation

ERIS-Rangfolge der Sequenzen der 1000 Generationen Optimierung. Gezeigt werden die zehn besten und die zehn schlechtesten Sequenzen. Die Rangfolge wurde mittels des $\Delta\Delta G$-Wertes erstellt. E_{diff} ist ein interner Wert von *ERIS*, $+/-$ die Ungenauigkeit.

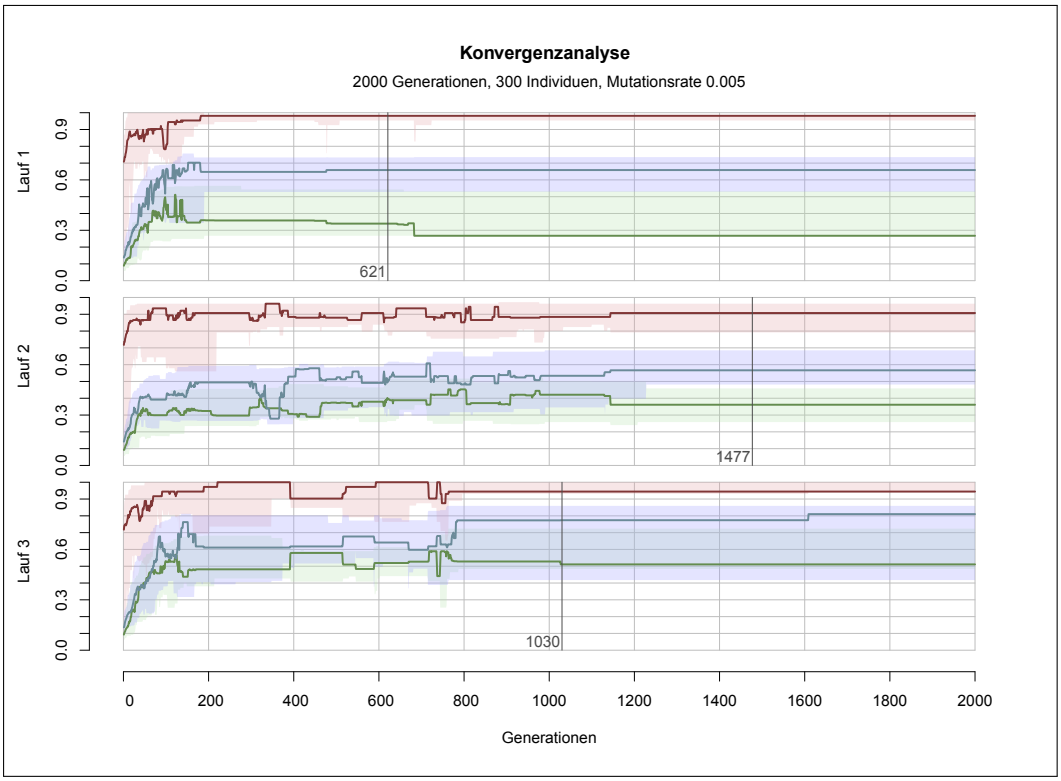
ID	Sequenz	E_{diff}	$+/-$	$\Delta\Delta G$
JW63	KFNKEQQNAFYEILHINSMNDHERNAFIQAMKDNNASARRVVATARERARAATT	21,45	3,30	2,47
JW25	KFNKEQQNAFYEILHINSMNDHHRNAMIQAMKKENASARRVIATARERARVTST	22,26	4,10	2,59
	KFNKEQQNAFYEILHINSMNNEHRNAFIQAMKDNNASARRVVATARERARVTAT	23,88	3,65	2,89
	KFNKEQQNAFYEILHINSMNDHHRNAMIQAMKKENASARRVIATARERARITST	22,64	2,57	2,92
	KFNKEQQNAFYEILHINSMNNEHRNAMIQAMKDNNASARRVVATARERARVSAT	25,15	4,35	3,31
	KFNKEQQNAFYEILHINSMNDHERNAFIQAMKDNNASARRVVATARERARVTST	23,51	4,39	4,30
JW2	KFNKEQQNAFYEILHIQSMNDHHRNAMIQAMKKENASARRVIATARERARITST	23,85	5,40	4,45
JW8	KFNKEQQNAFYEILHIDSMNDHHRNAMIQAMKKENASARRVIATARERARVTST	23,64	4,75	4,81
	KFNKEQQNAFYEILHINSMNDHERNAFIQAMKDNNSAARRAIATARERARAAST	22,44	1,34	5,16
	KFNKEQQNAFYEILHINSMNDHERNAFIQAMKDNNASARRAVATARERARASAT	22,45	3,23	5,22
	⋮			
	KFNKEQQNAFYEILHINSMNNEHRNAFIQAMKDNNSAARRAVATARERARVSST	32,62	2,52	14,31
	KFNKEQQNAFYEILHINSMNNHHRNAMIQAMKDNNSAARRAVATARERARVTST	36,13	2,27	15,04
	KFNKEQQNAFYEILHINSMNNEHRNAMIQAMKDNNSAARRAVATARERARVSST	34,88	2,42	15,14
JW26	KFNKEQQNAFYEILHINSMNDHHRNAMIQAMKKKKASLRRAVATIRERSRVATT	32,79	4,43	15,47
JW12	KFNKEQQNAFYEILHINSMNDHHRNAMIQAMKNKKASLRRAVATIRERSRVATT	35,72	7,28	16,46
	KFNKEQQNAFYEILEILDTNQEHRNAFIQAMKDESSAARRAVATARERARVTAT	32,56	3,30	16,77
JW18	KFNKEQQNAFYEILEIDSMNDHHRNAMIQAMKNKKASLRRAVATIRERSRVAST	34,63	6,05	17,56
	KFNKEQQNAFYEILHINSMNDHHRNAMIQAMKNKKASLRRAVATIRERSRVAST	39,07	5,64	20,39
JW55	KFNKEQQNAFYEILHINSMNNHHRNAMIQAMKNKKASLRRAVATIRERSRVATT	42,58	3,38	22,48
	KFNKEQQNAFYEILHINSMNDHHRNAMIQAMKKKKASLRRAVATIRERSRVAST	39,30	5,82	22,55

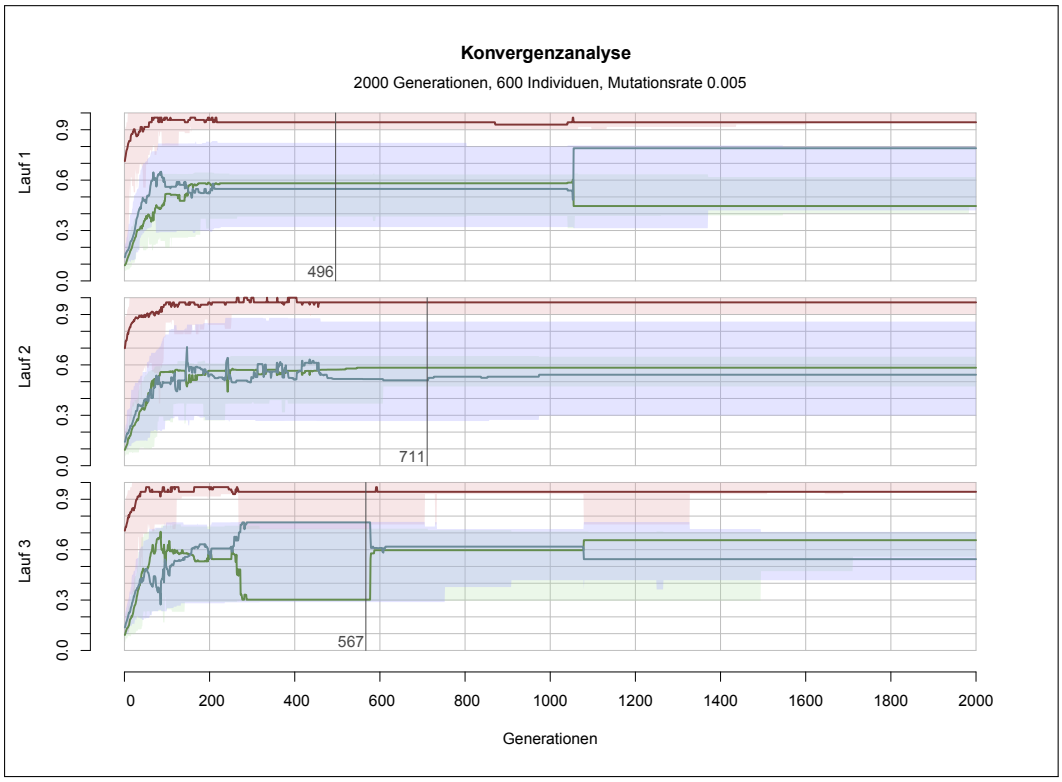
ERIS-Rangfolge der Sequenzen der 2000 Generationen-Optimierung. Gezeigt werden die zehn besten und die zehn schlechtesten Sequenzen. Die Rangfolge wurde mittels des $\Delta\Delta G$-Wertes erstellt. E_{diff} ist ein interner Wert von *ERIS*, $+/-$ die Ungenauigkeit.

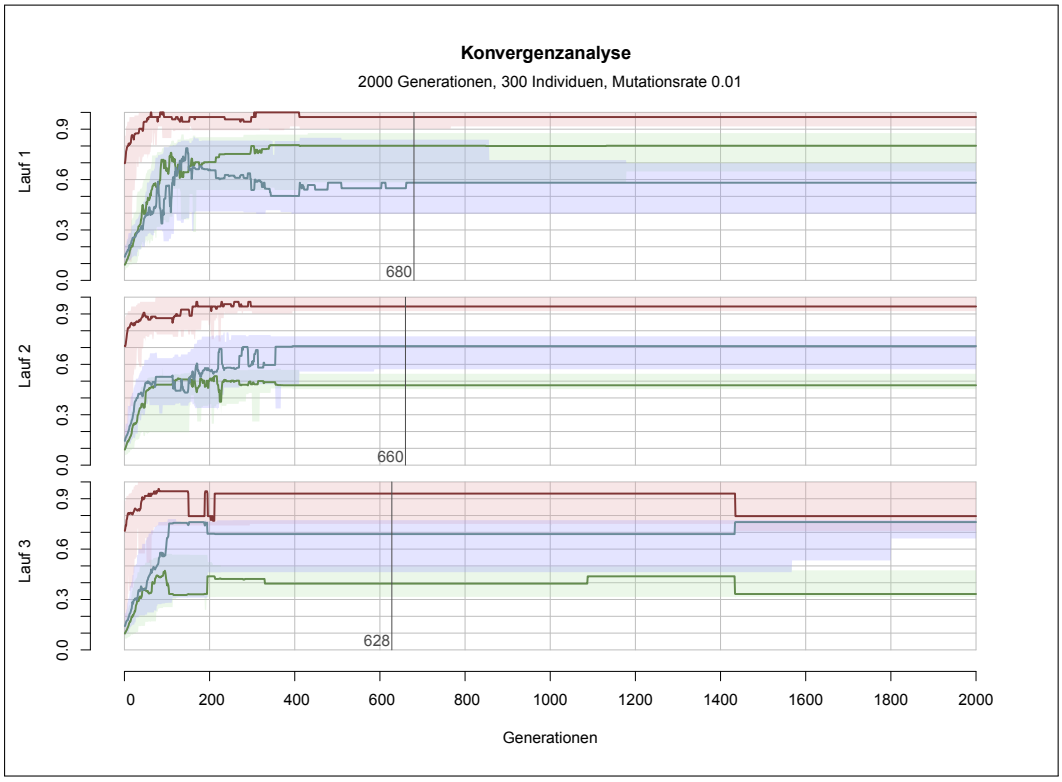
ID	Sequenz	E_{diff}	$+/-$	$\Delta\Delta G$
JW36	KFNKEQQNAFYEILHLVDTQQEQMNTFIQAVKRDSSAARRVAATARERARASSV	26,16	3,15	9,76
	KFNKEQQNAFYEILHLTTTHQDYQNTMIQAVKRDNSAARRVLATVRERARAAST	29,56	1,52	10,18
	KFNKEQQNAFYEILHLTTTEEDYQNTMIQAVKRDNSAARRIAATVRERSRVASV	30,22	2,90	11,02
JW19	KFNKEQQNAFYEILHLVDTQQEQQNTFIQAVKRDSSAARRVAATARERARASSV	24,86	2,65	11,20
	KFNKEQQNAFYEILHLTTTHQDYQNTMIQAVKRDNSAARRVAATARERARASVT	29,77	2,54	11,23
	KFNKEQQNAFYELLHITTTKEDYQNTMIQAVKRKNTAARRVLATIRERSRVAST	28,28	4,02	11,52
JW56	KFNKEQQNAFYEILHLTTTHQNYQNTMIQAVKRDNSAARRIAATARERARAASV	31,56	2,84	11,77
JW70	KFNKEQQNAFYEILHLTTTHQEQQNTFIQAVKRNNSAARRVAATARERARAASV	27,77	2,72	12,04
	KFNKEQQNAFYELLHITTTKEDYQNTMIQAVKRDNSAARRVLATIRERSRVAST	29,60	3,39	12,33
	KFNKEQQNAFYEILHLTTTHQNYQNTMIQAVKRDNSAARRIAATVRERARASVT	33,60	1,69	13,00
	⋮			
	KFNKEQQNAFYEILHLTTTHQQQQNTFIQAVKRNNSAARRVAATVRERSRASVT	36,53	3,38	20,54
	KFNKEQQNAFYEILHLTTTHQQHQNTFIQAVKHDNSAARRVAATARERARAAST	34,56	1,37	21,07
	KFNKEQQNAFYEILHLTTTHQQHQNTFIQAVKRNNSAARRVAATVRERARASVT	38,65	2,83	21,34
JW57	KFNKEQQNAFYEILHLTTTHQQHQNTFIQAVKRKSSAARRVAATARERSRASVT	34,00	3,22	21,40
	KFNKEQQNAFYEILHLTTTHQNYQNTMIQAVKRDNSAARRIAATIRERSRVAST	41,27	3,65	21,55
JW21	KFNKEQQNAFYEILHLTTTEEDYQNTMIQAVKRDNSAFRRVAATVRERSRVATY	43,50	2,17	22,06
JW72	KFNKEQQNAFYEILHLTTTHQHHQNTFIQAVKRDNSAARRVAATVRERARASVT	39,01	2,32	22,14
	KFNKEQQNAFYEILHLTTTHQQHQNTFIQAVKRNNSAARRVAATVRERSRASVT	39,48	1,69	23,10
	KFNKEQQNAFYEILHLTTTHQHHQNTFIQAVKRNNSAARRVAATVRERSRASVT	41,85	1,31	25,08
JW22	KFNKEQQNAFYEILHLTTTHQQHHNTFIQAVKRDNSAARRVAATARERARAAST	40,66	2,42	26,13

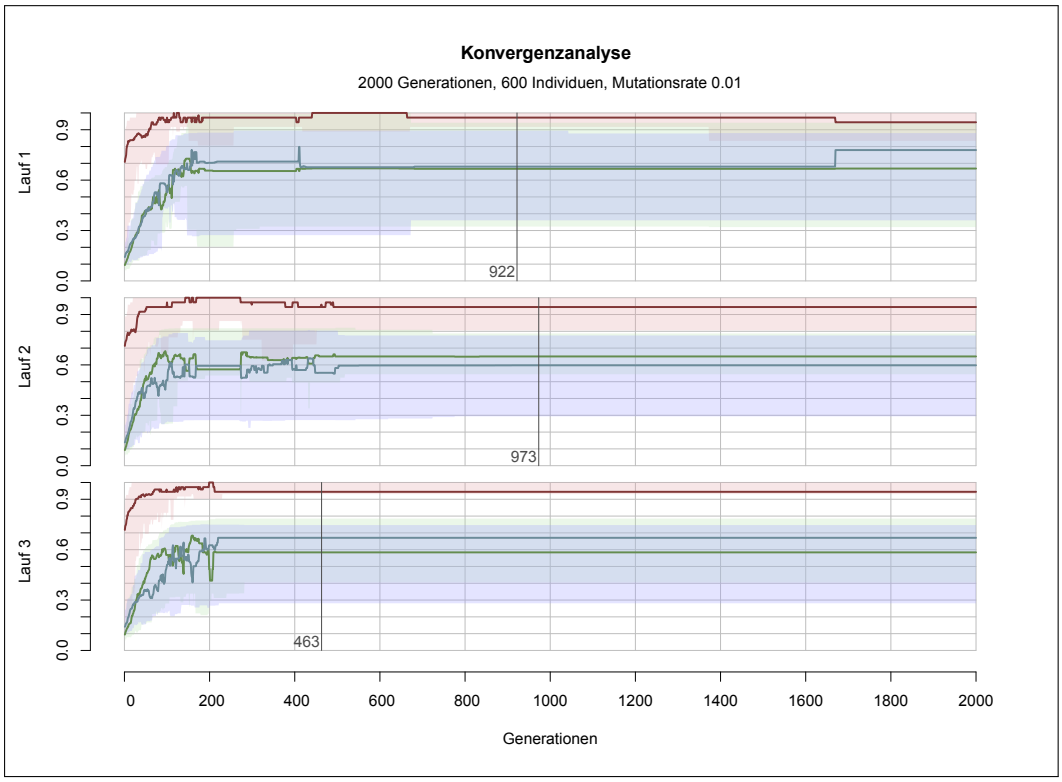
C Konvergenzgrafiken der Parameteroptimierung

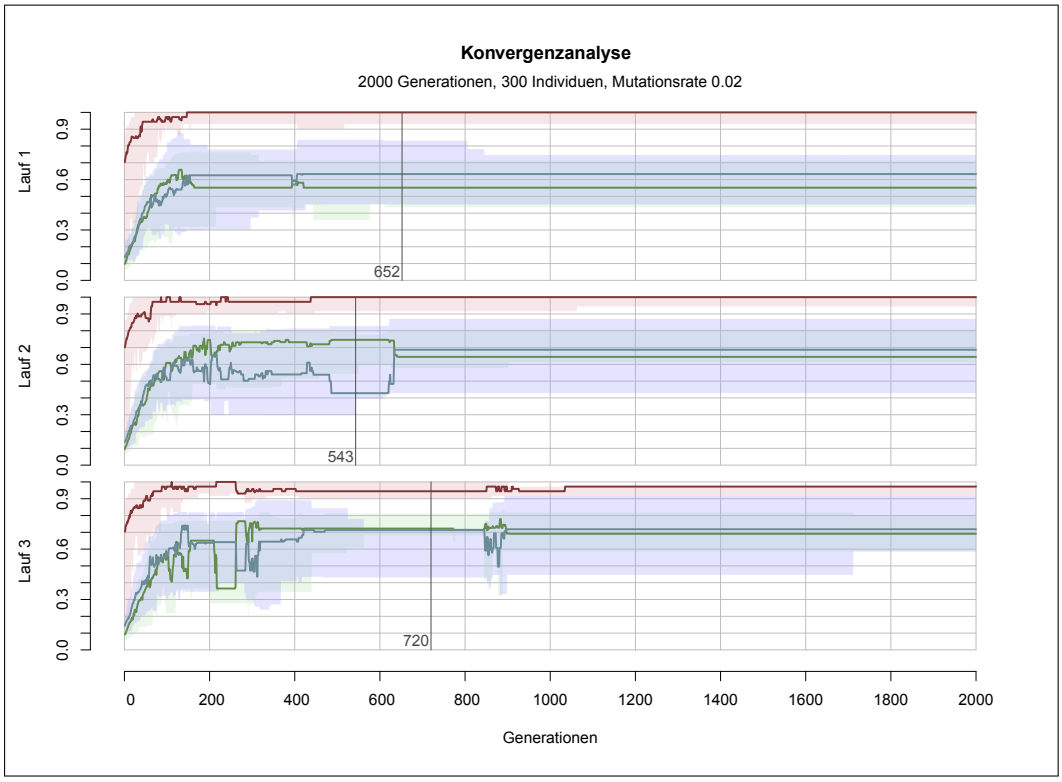
Konvergenzverhalten der GA-Optimierungen zur Parameteroptimierung an Hand der drei Fitnesswerte der Individuen auf der Pareto-Front einer jeden Generation. Die dargestellten Werte sind auf das jeweilige Maximum aller Simulationen normiert und auf den Bereich zwischen 0 und 1 skaliert. In Rot dargestellt ist der Sekundärstruktur-Wert, in Grün der Hydrophobizitäts- und in Blau der Molekulargewichts-Wert. Die farbige Linie gibt den mittleren Fitnesswert an, der farbige hellere Bereich markiert das Minimum und Maximum der Fitnesswerte.

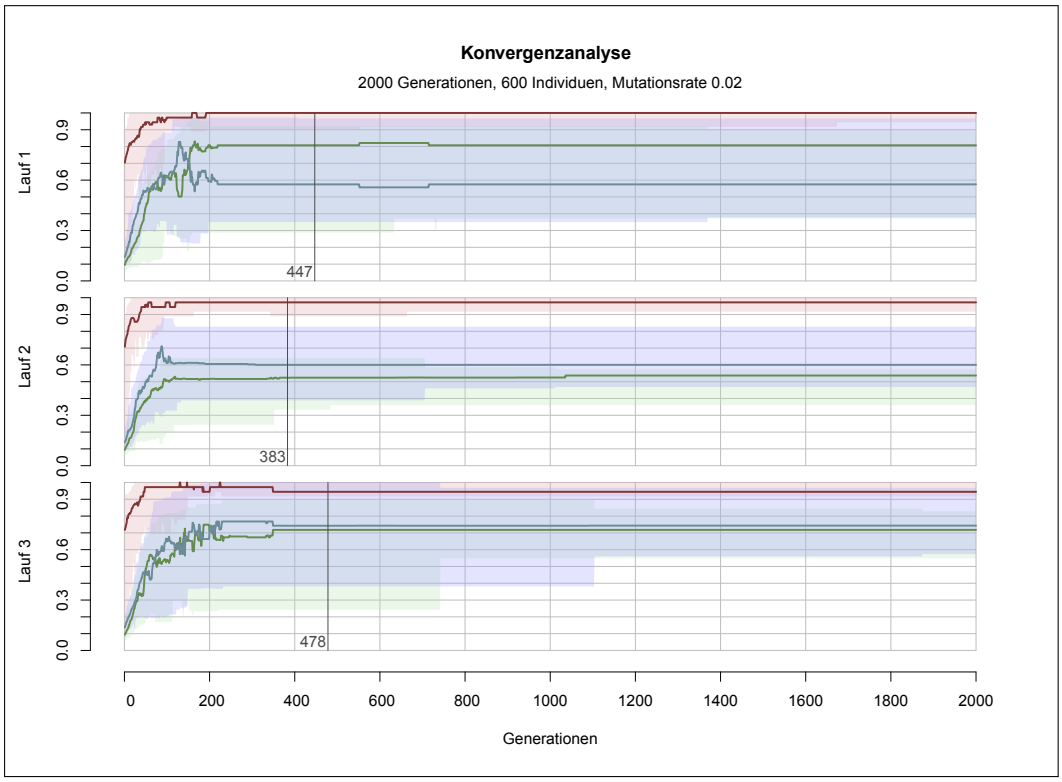



Konvergenzanalyse
2000 Generationen, 600 Individuen, Mutationsrate 0.005
Lauf 1
Lauf 2
Lauf 3
0.0
0.3
0.6
0.9
496
711
567
0
200
400
600
800
1000
1200
1400
1600
1800
2000
Generationen


Konvergenzanalyse
2000 Generationen, 300 Individuen, Mutationsrate 0.01
Lauf 1
Lauf 2
Lauf 3
0.0
0.3
0.6
0.9
680
660
628
0
200
400
600
800
1000
1200
1400
1600
1800
2000
Generationen

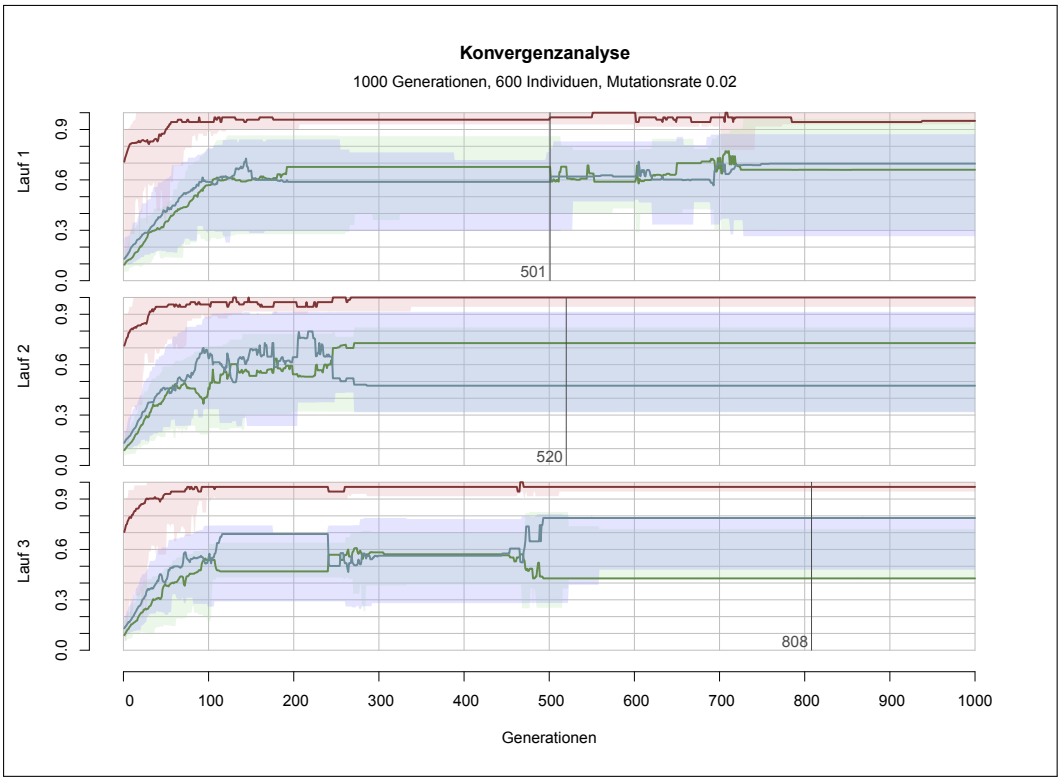

Konvergenzanalyse
2000 Generationen, 600 Individuen, Mutationsrate 0.01
Lauf 1
Lauf 2
Lauf 3
0.0
0.3
0.6
0.9
922
973
463
0
200
400
600
800
1000
1200
1400
1600
1800
2000
Generationen

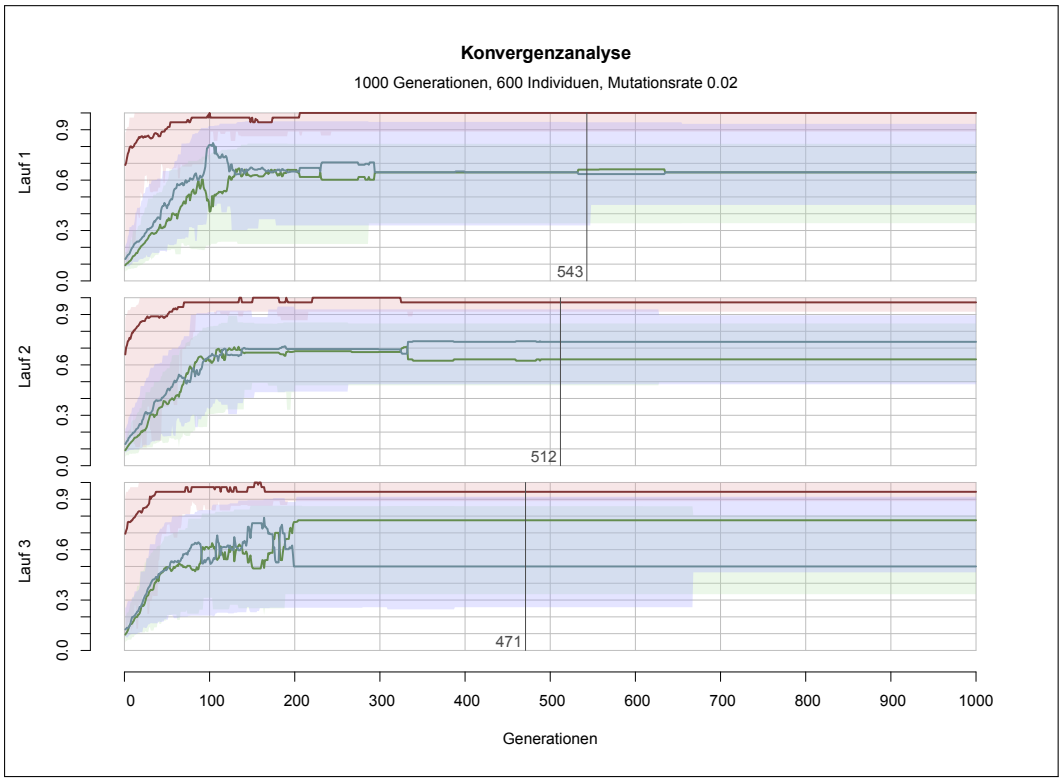

Konvergenzanalyse
2000 Generationen, 300 Individuen, Mutationsrate 0.02
Lauf 1
Lauf 2
Lauf 3
0.0
0.3
0.6
0.9
652
543
720
0
200
400
600
800
1000
1200
1400
1600
1800
2000
Generationen

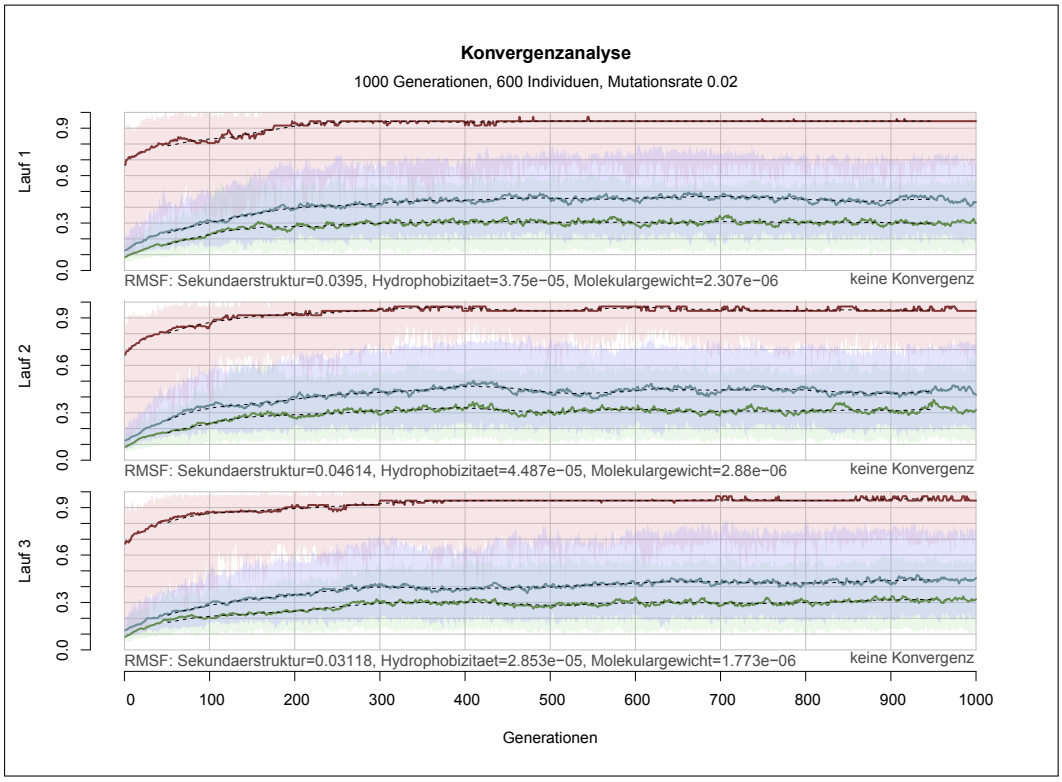

Konvergenzanalyse
2000 Generationen, 600 Individuen, Mutationsrate 0.02
Lauf 1
Lauf 2
Lauf 3
0.0
0.3
0.6
0.9
447
383
478
0
200
400
600
800
1000
1200
1400
1600
1800
2000
Generationen

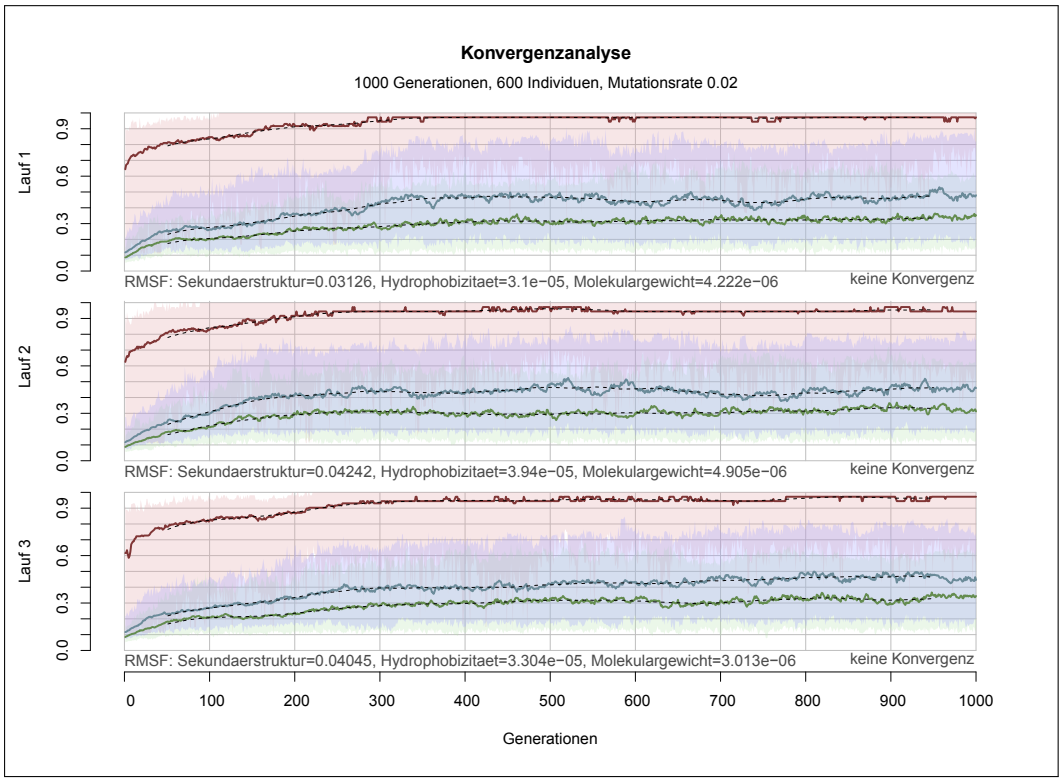
D Konvergenzgrafiken der Operatorenoptimierung

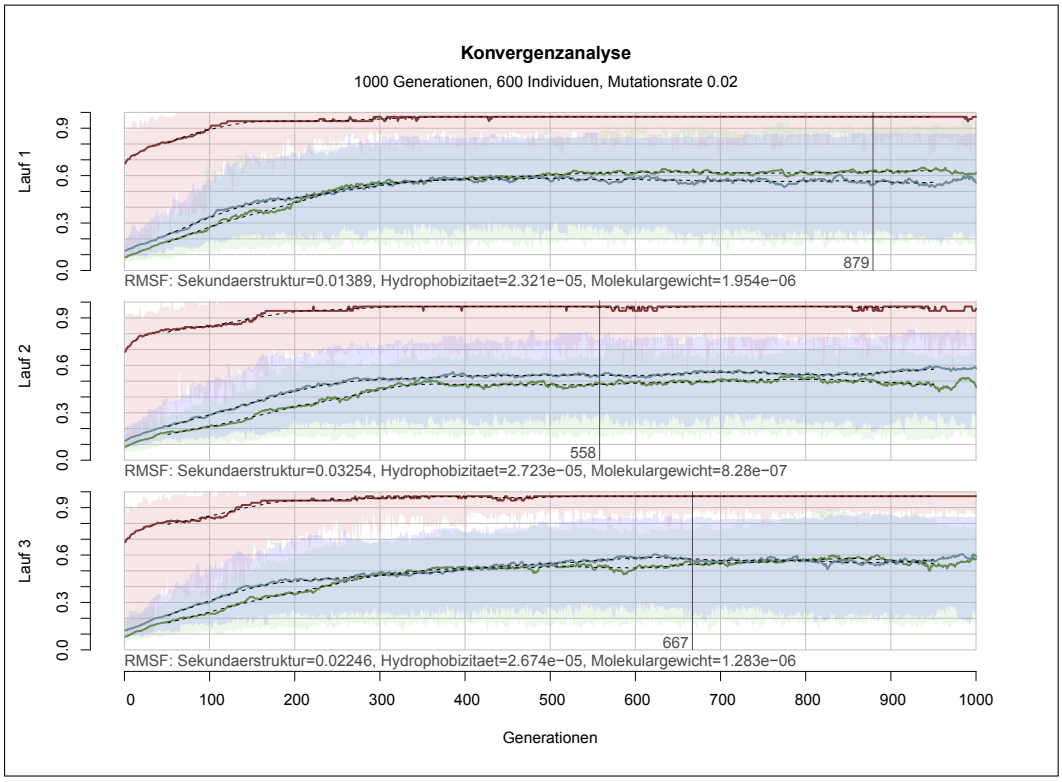
Konvergenzverhalten der GA-Optimierungen zur Operatorenoptimierung an Hand der drei Fitnesswerte der Individuen auf der Pareto-Front einer jeden Generation. Die dargestellten Werte sind auf das jeweilige Maximum aller Simulationen normiert und auf den Bereich zwischen 0 und 1 skaliert. In Rot dargestellt ist der Sekundärstruktur-Wert, in Grün der Hydrophobizitäts- und in Blau der Molekulargewichts-Wert. Die farbige Linie gibt den mittleren Fitnesswert an, der farbige hellere Bereich markiert das Minimum und Maximum der Fitnesswerte.

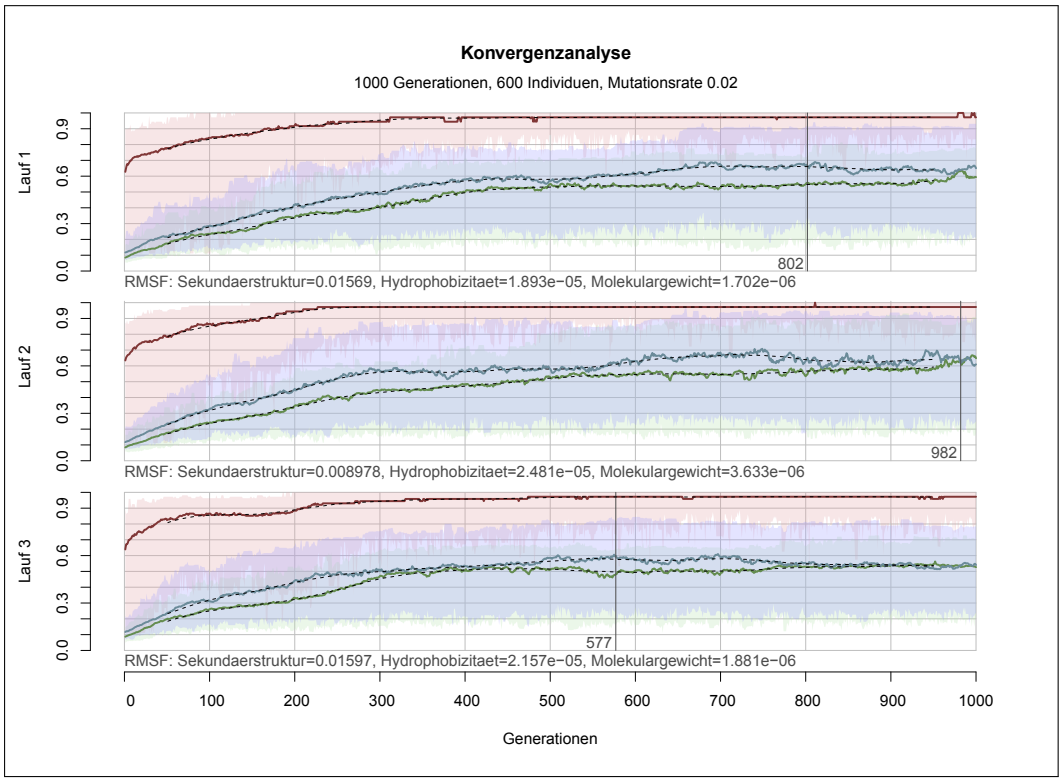



Konvergenzanalyse
1000 Generationen, 600 Individuen, Mutationsrate 0.02
Lauf 1
Lauf 2
Lauf 3
0.0
0.3
0.6
0.9
543
512
471
0
100
200
300
400
500
600
700
800
900
1000
Generationen


Konvergenzanalyse
1000 Generationen, 600 Individuen, Mutationsrate 0.02
Lauf 1
Lauf 2
Lauf 3
RMSF: Sekundaerstruktur=0.0395, Hydrophobizitaet=3.75e−05, Molekulargewicht=2.307e−06
keine Konvergenz
RMSF: Sekundaerstruktur=0.04614, Hydrophobizitaet=4.487e−05, Molekulargewicht=2.88e−06
keine Konvergenz
RMSF: Sekundaerstruktur=0.03118, Hydrophobizitaet=2.853e−05, Molekulargewicht=1.773e−06
keine Konvergenz
0
100
200
300
400
500
600
700
800
900
1000
Generationen


Konvergenzanalyse
1000 Generationen, 600 Individuen, Mutationsrate 0.02
Lauf 1
RMSF: Sekundaerstruktur=0.03126, Hydrophobizitaet=3.1e−05, Molekulargewicht=4.222e−06
keine Konvergenz
Lauf 2
RMSF: Sekundaerstruktur=0.04242, Hydrophobizitaet=3.94e−05, Molekulargewicht=4.905e−06
keine Konvergenz
Lauf 3
RMSF: Sekundaerstruktur=0.04045, Hydrophobizitaet=3.304e−05, Molekulargewicht=3.013e−06
keine Konvergenz
0.0 0.3 0.6 0.9
0 100 200 300 400 500 600 700 800 900 1000
Generationen


Konvergenzanalyse
1000 Generationen, 600 Individuen, Mutationsrate 0.02
Lauf 1
879
RMSF: Sekundaerstruktur=0.01389, Hydrophobizitaet=2.321e−05, Molekulargewicht=1.954e−06
Lauf 2
558
RMSF: Sekundaerstruktur=0.03254, Hydrophobizitaet=2.723e−05, Molekulargewicht=8.28e−07
Lauf 3
667
RMSF: Sekundaerstruktur=0.02246, Hydrophobizitaet=2.674e−05, Molekulargewicht=1.283e−06
0.0 0.3 0.6 0.9
0 100 200 300 400 500 600 700 800 900 1000
Generationen


Konvergenzanalyse
1000 Generationen, 600 Individuen, Mutationsrate 0.02
Lauf 1
Lauf 2
Lauf 3
0.0
0.3
0.6
0.9
802
982
577
RMSF: Sekundaerstruktur=0.01569, Hydrophobizitaet=1.893e−05, Molekulargewicht=1.702e−06
RMSF: Sekundaerstruktur=0.008978, Hydrophobizitaet=2.481e−05, Molekulargewicht=3.633e−06
RMSF: Sekundaerstruktur=0.01597, Hydrophobizitaet=2.157e−05, Molekulargewicht=1.881e−06
0
100
200
300
400
500
600
700
800
900
1000
Generationen

E ERIS-Rangfolge der optimierten GA-Simulationen

ERIS-Rangfolge der jeweils fünf am besten bewerteten Sequenzen der optimierten GA-Simulationen. Die Rangfolge wurde mittels des $\Delta\Delta G$-Wertes erstellt. Die obere Tabelle zeigt die Ergebnisse der drei Läufe mit der Molekulargewichts-Fitnessfunktion (m1–m3), die untere Tabelle die Läufe mit der Epitopsy-Fitnessfunktion (e1–e3).

ID	Sequenz	E_{diff}	+/−	$\Delta\Delta G$
JW138m1	KFNEKQQNAFYEILHLSKLNEQERNAFIQSLKKNSSSARRVIATARERARVTTT	7,71	3,77	-3,29
JW107m1	KFDQKQQNAFYEILHLSKLNEQQRNAFIQSLKKNSSSFRRVVATARERARVTSV	11,34	4,18	-1,19
JW53m1	KFNEKQQNAFYEILHLSKLNQQERNAFIQSLKKNSSSFRRVVATARERARVTTT	10,96	4,24	-1,1
JW82m1	EFNEKQQNAFYEILHLSKLNQQERNAFIQSLKKNSSSFRRVVATARERARVSSA	12,33	2,8	-0,17
JW114m1	EFNEKQQNAFYEILELSKLNEQQRNAFIQSLKKNSSSFRRVVATARERARVTTT	12,44	5,77	-0,1
JW51m2	KFNKEQQNAFYEILHISNMNNHEHNTLIQAMKRNSSTARRIVATARERARVTAT	23,68	2,92	3
JW20m2	EFNKEQQNAFYEILHISNMNNHEHNTLIQAMKRNSSSARRVVATARERARVTAD	24,19	3,21	3,64
JW18m2	KFNKEQQNAFYEILHISNMNNHEHNTLIQAMKRNSSSARRVVATARERARVTAT	23,85	4,1	3,8
JW7m2	RFNKEQQNAFYEILHLTNMSHKERNTLIQAMKKNSSAARRAVATARERARASAS	19,78	2,02	4,06
JW3m2	KFNKQQQNAFYELLHITNMSEKERNTLIQAMKRSSSAARRAVATARERARASAS	19,28	1,88	4,62
JW106m3	KFNKEQQNAFYEILHLSKLNEQERNAFIQAMKRDSSSFRRVVATARERARVSAD	13,93	6,9	-0,84
JW35m3	KFNKEQQNAFYQILELSKLNEQERNAFIQAMKKNSSSARRVVATARERARATAT	12,25	3,44	0,25
JW1m3	KFDKEQQNAFYEILHLSKLNEEERNAFIQAMKKKSSSARRVVATARERARVSTA	10,88	3,54	1,07
JW83m3	KFNKEQQNAFYEILHLSKLNEQERNAFIQAMKRNSSSARRVVATARERARVTAT	17,3	1,96	1,09
JW48m3	KFNKEQQNAFYQILELSKLNEQERNAFIQAMKRDSSSFRRVVATARERARVSAT	16,26	4,07	1,13
JW15e1	EFHRHQQNMFYQVLHLTNLREDNQNAMIQSIKNNHTSLRRLLATLRERARVTTT	25,53	2,24	-1,11
JW4e1	EFHRHQQNMFYEVLHLTNLREDEENAMIQSIKNDHTSLRRLLATLRERARVTYM	28,56	2,99	-0,8
JW16e1	EFHRHQQNMFYQVLHLTNLREDDENAMIQSIKNNHTSLRRLLATLRERARVTYA	28,41	3,78	0,14
JW3e1	DFHRDQQNMFYNILHFTNLEENHRNAMIQSIKNNHTSLRRVLATLRERARVTYS	29,86	6,19	1,23
JW1e1	NFHRDQQNAFYEILHFMNSTDHHRNAMIQAMKDDHDSMRRILATLRERARVTIN	31,94	5,29	3,49
JW1e2	TFKREQQNAFYELLYFTNMNRTHRNSFIQAFKDNETTIRRVFATLRERMRITTV	29,91	3,99	0,81
JW3e2	HFHKEQQNAFYELLYFTNAHRSQRNTLIQAFKDDETTIRRVFATLRERMRVTTV	27,58	5,02	2,23
JW2e2	HFHREQQNAFYQLLYFTNAHRTQRNTFIQAFKDNQTTIRRVFATLRERQRITTV	30,91	5,67	3,52
JW6e2	NFEKEQQNAFYKLLYFTNMHRSQRNSLIQAFKKDETTIRRVFATLRERARVTTT	27,49	4,11	7,42
JW12e2	QFEKDQQNAFYELLYFTNAHRSQRNSLIQAMKNDHSTIRRLFATLRERARVTTT	29,62	4,76	7,71
JW23e3	NFHRHQQNMFYHVLHLYNLNEEERNDFIQSLKDKSSSFRRLLATFRERARVTTT	24,87	6,68	2,09
JW13e3	NFHRHQQNMFYEVLHLYDLNNEDRNAFIQTLKRNTSSFRRLIATFRERARVTTT	30,64	6,73	2,78
JW20e3	KFHRHQQNMFYHVLHLYNLNNEERNAFIQTLKDNTSSFRRLIATFRERARVTTT	29,03	6,46	2,91
JW1e3	QFERHQQNMFYHVLHLYNLNDEQRNAFIQSLKENNANLRRVIATFRERARVTTT	31,91	4,38	3,72
JW15e3	KFHRHQQNMFYEVLHLYNLDNEERNAFIQSLKRNTSSFRRLIATFRERARVTTN	31,2	9,85	5,63

Literaturverzeichnis

[1] Abromand, A. ; Bayro, T. ; Bosse, D. ; Budeus, B. ; Erdem, E. ; Grotegerd, D. ; Kruse, T. ; Riemenschneider, M.: *Evolutionäre Algorithmen zur Generierung ähnlicher Proteinsequenzen*, Westfälische Wilhelms-Universität Münster, Diplomarbeit, Mar 2009

[2] Alexander, P. A. ; He, Y. ; Chen, Y. ; Orban, J. ; Bryan, P. N.: The design and characterization of two proteins with 88identity but different structure and function. In: *Proc Natl Acad Sci U S A* 104 (2007), Jul, Nr. 29, S. 11963–11968

[3] Armano, G. ; Ledda, F. ; Vargiu, E.: Sum-Linear Blosum: A Novel Protein Encoding Method for Secondary Structure Prediction. In: *COSIWN* 6 (2009), S. 71–77

[4] Bairoch, A. ; Apweiler, R.: The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000. In: *Nucleic Acids Res* 28 (2000), Jan, Nr. 1, S. 45–48

[5] Baker, N. A. ; Sept, D. ; Joseph, S. ; Holst, M. J. ; McCammon, J. A.: Electrostatics of nanosystems: application to microtubules and the ribosome. In: *Proc. Natl. Acad. Sci. U.S.A.* 98 (2001), Aug, Nr. 18, S. 10037–10041

[6] Bernstein, F. C. ; Koetzle, T. F. ; Williams, G. J. ; Meyer, E. F. ; Brice, M. D. ; Rodgers, J. R. ; Kennard, O. ; Shimanouchi, T. ; Tasumi, M.: The Protein Data Bank. A computer-based archival file for macromolecular structures. In: *Eur J Biochem* 80 (1977), November, Nr. 2, S. 319–324

[7] Blackwell, T. K. ; Weintraub, H.: Differences and similarities in DNA-binding preferences of MyoD and E2A protein complexes revealed by binding site selection. In: *Science* 250 (1990), November, Nr. 4984, S. 1104–1110

[8] Case, D. A. ; Darden, T. A. ; Cheatham, T. E. ; Simmerling, C. L. ; Wang, J. ; Duke, R. E. ; Luo, R. ; Crowley, M. ; Walker, R. C. ; Zhang, W. ; Merz, K. M. ; Wang, B. ; Hayik, S. ; Roitberg, A. ; Seabra, G. ; Kolossvary, I. ; Wong, K. F. ; Paesani, F. ; Vanicek, J. ; Wu, X. ; Brozell, S. R. ; Steinbrecher, T. ; Gohlke, H. ; Yang, L. ; Tan, C. ; Mongan, J. ; Hornak, V. ; Cui, G. ; Mathews, D. H. ; Seetin, M. G. ; Sagui, C. ; Babin, V. ; Kollman, P. A.: *Amber 10*. University of California, 2008

[9] Cedergren, L. ; Andersson, R. ; Jansson, B. ; Uhlén, M. ; Nilsson, B.: Mutational analysis of the interaction between staphylococcal protein A and human IgG1. In: *Protein Eng* 6 (1993), Nr. 4, S. 441–448

[10] Chen, L. ; DeVries, A. L. ; Cheng, C. H.: Convergent evolution of antifreeze glycoproteins in Antarctic notothenioid fish and Arctic cod. In: *Proc Natl Acad Sci U S A* 94 (1997), Nr. 8, S. 3817–3822

[11] Crooks, G. E. ; Hon, G. ; Chandonia, J. M. ; Brenner, S. E.: WebLogo: a sequence logo generator. In: *Genome Res* 14 (2004), Jun, Nr. 6, S. 1188–90. – URL http://weblogo.berkeley.edu

[12] Darwin, C.: *On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life*. J. Murray, 1859

[13] Darwin, C. ; Wallace, A. R.: On the Tendency of Species to form Varieties; and on the Perpetuation of Varieties and Species by Natural Means of Selection. In: *Journal of the Proceedings of the Linnean Society of London* Zoology 3 (1858), S. 46––50

[14] Deb, K.: *Multi-Objective Optimization Using Evolutionary Algorithms*. 1. Wiley, 2009. – ISBN 978-0470743614

[15] Deb, K. ; Pratap, A. ; Agarwal, S. ; Meyarivan, T.: A fast and elitist multiobjective genetic algorithm : NSGA-II. In: *Evolutionary Computation, IEEE Transactions on* 6 (2002), Nr. 2, S. 182–197

[16] Deisenhofer, J.: Crystallographic refinement and atomic models of a human Fc fragment and its complex with fragment B of protein A from Staphylococcus aureus at 2.9- and 2.8-A resolution. In: *Biochemistry* 20 (1981), Apr, Nr. 9, S. 2361–2370

[17] Desjarlais, J. R. ; Handel, T. M.: De novo design of the hydrophobic cores of proteins. In: *Protein Sci.* 4 (1995), Oct, S. 2006–2018

[18] Devos, D. ; Valencia, A.: Practical limits of function prediction. In: *Proteins* 41 (2000), Oct, Nr. 1, S. 98–107

[19] Ding, F. ; Dokholyan, N. V.: Emergence of protein fold families through rational design. In: *PLoS Comput Biol* 2 (2006), Jul, Nr. 6

[20] Dolinsky, T. J. ; Nielsen, J. E. ; McCammon, J. A. ; Baker, N. A.: PDB2PQR: an automated pipeline for the setup of Poisson-Boltzmann electrostatics calculations. In: *Nucleic Acids Res.* 32 (2004), Jul, Nr. Web Server issue, S. W665–667

[21] Dybowski, J. N.: *Development of a method for optimal superposition of pairs of similar macromolecules*, Fachhochschule Bingen, Bingen University of Applied Sciences, Diplomarbeit, Nov 2006

[22] Dybowski, J. N. ; Heider, D. ; Hoffmann, D.: Prediction of co-receptor usage of HIV-1 from genotype. In: *PLoS Comput Biol* 6 (2010), Apr, Nr. 4, S. e1000743

[23] Dybowski, J. N. ; Heider, D. ; Hoffmann, D.: Structure of HIV-1 quasispecies as early indicator for switches of co-receptor tropism. In: *AIDS Res Ther* 7 (2010), Nov, Nr. 1, S. 41

[24] Dybowski, J. N. ; Riemenschneider, M. ; Hauke, S. ; Pyka, M. ; Verheyen, J. ; Hoffmann, D. ; Heider, D.: Improved Bevirimat resistance prediction by combination of structural and sequence-based classifiers. In: *BioData Min* 4 (2011), S. 26

[25] Fasman, G. D.: *Handbook of Biochemistry: Section D Physical Chemical Data, Volume I.* CRC Press, 1976. – ISBN 0878195092

[26] Fogolari, F. ; Brigo, A. ; Molinari, H.: The Poisson-Boltzmann equation for biomolecular electrostatics: a tool for structural biology. In: *J. Mol. Recognit.* 15 (2002), Nr. 6, S. 377–392

[27] Fonseca, C. M. ; Fleming, P. J.: Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In: *Genetic Algorithms: Proceedings of the Fifth International Conference*, Morgan Kaufmann, 1993, S. 416–423

[28] Forsgren, A. ; Sjöquist, J.: "Protein A"from S. aureus. I. Pseudo-immune reaction with human gamma-globulin. In: *J Immunol* 97 (1966), Dec, Nr. 6, S. 822–827

[29] Forst, W. ; Hoffmann, D.: *Optimization - Theory and Practice.* Springer, 2010. – ISBN 978-0387789767

[30] Fromer, M. ; Yanover, C.: Accurate prediction for atomic-level protein design and its application in diversifying the near-optimal sequence space. In: *Proteins* 75 (2009), May, Nr. 3, S. 682–705

[31] Gilbert, D. G.: *Phylodendron.* Online. – URL http://iubio.bio.indiana.edu/treeapp/treeprint-form.html

[32] Gotoh, O.: An improved algorithm for matching biological sequences. In: *J Mol Biol* 162 (1982), Dec, Nr. 3, S. 705–708

[33] GROMACS Documentation: *GROMOS force fields.* Online. – URL http://www.gromacs.org/Documentation/Terminology/Force_Fields/GROMOS/

[34] Gronwald, W. ; Hohm, T. ; Hoffmann, D.: Evolutionary Pareto-optimization of stably folding peptides. In: *BMC Bioinformatics* 9 (2008), S. 109

[35] Guerois, R. ; Nielsen, J. E. ; Serrano, L.: Predicting changes in the stability of proteins and protein complexes: a study of more than 1000 mutations. In: *BMC Bioinformatics* 320 (2002), Jul, Nr. 2, S. 369–387

[36] Guerrero, J. L. ; Martí, L. ; Berlanga, A. ; García, J. ; López, J. M. M.: Introducing a robust and efficient stopping criterion for MOEAs. In: *IEEE Congress on Evolutionary Computation*, IEEE, 2010, S. 1–8

[37] Heider, D. ; Appelmann, J. ; Bayro, T. ; Dreckmann, W. ; Held, A. ; Winkler, J. ; Barnekow, A. ; Borschbach, M.: A computational approach for the identification of small GTPases based on preprocessed amino acid sequences. In: *Technol Cancer Res Treat* 8 (2009), Oct, Nr. 5, S. 333–341

[38] Heider, D. ; Hauke, S. ; Pyka, M. ; Kessler, D.: Insights into the classification of small GTPases. In: *Adv Appl Bioinform Chem* 3 (2010), S. 15–24

[39] Hess, B. ; Kutzner, C. ; Spoel et al., D. van der: GROMACS 4: Algorithms for Highly Efficient, Load-Balanced, and Scalable Molecular Simulation. In: *J Chem Theory Comput* 4 (2008), Nr. 3, S. 435–447

[40] Högbom, M. ; Eklund, M. ; Nygren, P. ; Norldlund, P.: Structural basis for recognition by an in vitro evolved affibody. In: *Proc Natl Acad Sci U S A* 100 (2003), Nr. 6, S. 3191–3196

[41] Holland, J. H.: *Adaptation in Natural and Artificial Systems*. Reprint. The Mit Press, 1992. – ISBN 978-0262581110

[42] Horn, J. ; Nafpliotis, N. ; Goldberg, D. E.: A Niched Pareto Genetic Algorithm for Multiobjective Optimization. In: *In Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*, 1994, S. 82–87

[43] Huber, G. A. ; McCammon, J. A.: Browndye: A Software Package for Brownian Dynamics. In: *Comput Phys Commun* 181 (2010), Nov, Nr. 1, S. 1896–1905.

[44] Jones, D. T.: De novo protein design using pairwise potentials and a genetic algorithm. In: *Protein Sci.* 3 (1994), Apr, S. 567–574

[45] Jones, D. T.: Protein secondary structure prediction based on position-specific scoring matrices. In: *J. Mol. Biol.* 292 (1999), Sep, S. 195–202

[46] Kaplan, J. ; DeGrado, W. F.: De novo design of catalytic proteins. In: *Proc Natl Acad Sci U S A* 101 (2004), Aug, Nr. 32, S. 11566–11570

[47] Knowles, J. ; Thiele, L. ; Zitzler, E.: A Tutorial on the Performance Assessment of Stochastic Multiobjective Optimizers / Computer Engineering and Networks Laboratory (TIK), ETH Zurich, Switzerland. 2006. – Forschungsbericht. revised version

[48] Koppensteiner, W. A. ; Lackner, P. ; Wiederstein, M. ; Sippl, M. J.: Characterization of novel proteins based on known protein structures. In: *J Mol Biol* 296 (2000), Mar, Nr. 4, S. 1139–1152

[49] Kyte, J. ; Doolittle, R. F.: A simple method for displaying the hydropathic character of a protein. In: *J Mol Biol* 157 (1982), May, Nr. 1, S. 105–132

[50] Larkin, M. A. ; Blackshields, G. ; Brown, N. P. ; Chenna, R. ; McGettigan, P. A. ; McWilliam, H. ; Valentin, F. ; Wallace, I. M. ; Wilm, A. ; Lopez, R. ; Thompson, J. D. ; Gibson, T. J. ; Higgins, D. G.: Clustal W and Clustal X version 2.0. In: *Bioinformatics (Oxford, England)* 23 (2007), November, Nr. 21, S. 2947–2948. – URL http://www.ebi.ac.uk/Tools/msa/clustalw2/

[51] Lazar, G. A. ; Dang, W. ; Karki, S. ; Vafa, O. ; Peng, J. S. ; Hyun, L. ; Chan, C. ; Chung, H. S. ; Eivazi, A. ; Yoder, S. C. ; Vielmetter, J. ; Carmichael, D. F. ; Hayes, R. J. ; Dahiyat, B. I.: Engineered antibody Fc variants with enhanced effector function. In: *Proc Natl Acad Sci U S A* 103 (2006), Mar, Nr. 11, S. 4005–4010

[52] Levy, S. ; Sutton, G. ; Ng, P. C. ; Feuk, L. ; Halpern, A. L. ; Walenz, B. P. ; Axelrod, N. ; Huang, J. ; Kirkness, E. F. ; Denisov, G. ; Lin, Y. ; MacDonald, J. R. ; Pang, A. W. ; Shago, M. ; Stockwell, T. B. ; Tsiamouri, A. ; Bafna, V. ; Bansal, V. ; Kravitz, S. A. ; Busam, D. A. ; Beeson, K. Y. ; McIntosh, T. C. ; Remington, K. A. ; Abril, J. F. ; Gill, J. ; Borman, J. ; Rogers, Y. H. ; Frazier, M. E. ; Scherer, S. W. ; Strausberg, R. L. ; Venter, J. C.: The diploid genome sequence of an individual human. In: *PLoS Biol.* 5 (2007), Sep, Nr. 10, S. e254

[53] Lippe, W.-M.: *Soft-Computing*. Springer Berlin Heidelberg, 2005. – ISBN 978-3540209720

[54] Ljungberg, U. K. ; Jansson, B. ; Niss, U. ; Nilsson, R. ; Sandberg, B. E. ; Nilsson, B.: The interaction between different domains of staphylococcal protein A and human polyclonal IgG, IgA, IgM and F(ab')2: separation of affinity from specificity. In: *Mol Immunol* 30 (1993), Oct, Nr. 14, S. 1279–1285

[55] Lobo, F. G. ; Lima, C. F. ; Martires, H.: An architecture for massive parallelization of the compact genetic algorithm. In: *CoRR* cs.NE/0402049 (2004)

[56] Ma, P. C. ; Rould, M. A. ; et al., H. W.: Crystal structure of MyoD bHLH domain-DNA complex: perspectives on DNA recognition and implications for transcriptional activation. In: *Cell* 77 (1994), Nr. 3, S. 451–9

[57] MacKerell, A. D. ; Banavali, N. ; Foloppe, N.: Development and current status of the CHARMM force field for nucleic acids. In: *Biopolymers* 56 (2000), Nr. 4, S. 257–265

[58] Martí, L. ; García, J. ; Berlanga, A. ; Molina, J. M.: A cumulative evidential stopping criterion for multiobjective optimization evolutionary algorithms. In: *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, ACM, 2007 (GECCO '07), S. 911–911. – ISBN 978-1-59593-697-4

[59] MATLAB: *Version 7.13.0.564 (R2010a), 64-bit*. Natick, Massachusetts : The MathWorks Inc., Aug 2011

[60] Michalewicz, Z. ; Krawczyk, J. B. ; Kazemi, M. ; Janikow, C. Z.: Genetic algorithms and optimal control problems. In: *Decision and Control, 1990., Proceedings of the 29th IEEE Conference on* Bd. 3, dec 1990, S. 1664–1666

[61] Moorhead, G. B. G. ; Wever, V. D. ; Templeton, G. ; Kerk, D.: Evolution of protein phosphatases in plants and animals. In: *Biochem J* 417 (2009), Nr. 2, S. 401–409

[62] Moustafa, A.: *JAligner: Java implementation of the Smith-Waterman algorithm for biological sequence alignment.* Mai 2005. – URL http://jaligner.sourceforge.net/

[63] Nilsson, B. ; Moks, T. ; Jansson, B. ; Abrahmsén, L. ; Elmblad, A. ; Holmgren, E. ; Henrichson, C. ; Jones, T. A. ; Uhlén, M.: A synthetic IgG-binding domain based on staphylococcal protein A. In: *Protein Eng* 1 (1987), Feb-Mar, Nr. 2, S. 107–113

[64] Northrup, S. H. ; Allison, S. A. ; McCammon, J. A.: Brownian dynamics simulation of diffusion-influenced bimolecular reactions. In: *J. Chem. Phys* 80 (1984), Nr. 4, S. 1517—-1524

[65] Ong, S. ; Lin, H. ; Chen, Y. ; Li, Z. ; Cao, Z.: Efficacy of different protein descriptors in predicting protein functional families. In: *BMC Bioinformatics* 8 (2007), Nr. 1, S. 300

[66] Oracle Corporation: *Java SE Development Kit 6 Update 31.* Online. – URL http://www.oracle.com/technetwork/java/javase/downloads/jdk-6u31-download-1501634.html

[67] Ratliff, T. L. ; McCool, R. E. ; Catalona, W. J.: Interferon induction and augmentation of natural-killer activity by Staphylococcus protein A. In: *Cell Immunol* 57 (1981), Jan, Nr. 1, S. 1–12

[68] Roberts, K. E. ; Cushing, P. R. ; Boisguerin, P. ; Madden, D. R. ; Donald, B. R.: Computational Design of a PDZ Domain Peptide Inhibitor that Rescues CFTR Activity. In: *PLoS Comput. Biol.* 8 (2012), Apr, Nr. 4, S. e1002477

[69] Romberg, W.: Vereinfachte numerische Integration. In: *Det Kongelige Norske Videnskabers Selskab Forhandlinger (Trondheim)* 28(7) (1955), S. 7

[70] Röthlisberger, D. ; Khersonsky, O. ; Wollacott, A. M. ; Jiang, L. ; DeChancie, J. ; Betker, J. ; Gallaher, J. L. ; Althoff, E. A. ; Zanghellini, A. ; Dym, O. ; Albeck, S. ; Houk, K. N. ; Tawfik, D. S. ; Baker, D.: Kemp elimination catalysts by computational enzyme design. In: *Nature* 453 (2008), May, Nr. 7192, S. 190–195

[71] Sali, A. ; Blundell, T. L.: Comparative protein modelling by satisfaction of spatial restraints. In: *J. Mol. Biol.* 234 (1993), Dec, Nr. 3, S. 779–815

[72] Schilling, S. ; Wasternack, C. ; Demuth, H.-U.: Glutaminyl cyclases from animals and plants: a case of functionally convergent protein evolution. In: *Biol Chem* 389 (2008), Nr. 8, S. 983–991

[73] Schrödinger, LLC: *The PyMOL Molecular Graphics System, Version 1.3r1.* August 2010. – URL http://www.pymol.org

[74] Scott, L. P. B. ; Chahine, J. ; Ruggiero, J. R.: Using genetic algorithm to design protein sequence. In: *Applied Mathematics and Computation* 200 (2008), Nr. 1, S. 1–9

[75] Scott, W. R. P. ; Hunenberger, P. H. ; Tironi, I. G. ; Mark, A. E. ; Billeter, S. R. ; Fennen, J. ; Torda, A. E. ; Huber, T. ; Kruger, P. ; Gunsteren, W. F. van: The GROMOS Biomolecular Simulation Program Package. In: *J Phys Chem* 103 (1999), S. 3596–3607

[76] Shifman, J. M. ; Choi, M. H. ; Mihalas, S. ; Mayo, S. L. ; Kennedy, M. B.: Ca2+/calmodulin-dependent protein kinase II (CaMKII) is activated by calmodulin with two bound calciums. In: *Proc Natl Acad Sci U S A* 103 (2006), Sep, Nr. 38, S. 13968–13973

[77] Sjödahl, J. ; Möller, G.: The Fc Binding Regions in Protein A are not Responsible for the Polyclonal B Cell Activating Property of Protein A. In: *Scandinavian Journal of Immunology* 10 (1979), Nr. 6, S. 593–596

[78] Smith, T. F. ; Waterman, M. S.: Identification of common molecular subsequences. In: *J Mol Biol* 147 (1981), Mär, Nr. 1, S. 195–197

[79] Srinivas, N. ; Deb, K: Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. In: *Evolutionary Computation* 2 (1994), S. 221–248

[80] Stoer, J.: *Numerische Mathematik 1: Eine Einführung - unter Berücksichtigung von Vorlesungen von F.L. Bauer (Springer-Lehrbuch) (v. 1) (German Edition).* Springer, 2004. – ISBN 3540213953

[81] Suarez, M. ; Tortosa, P. ; Garcia-Mira, M. M. ; Rodríguez-Larrea, Da. ; Godoy-Ruiz, R. ; Ibarra-Molero, B. ; Sanchez-Ruiz, J. M. ; Jaramillo, A.: Using multi-objective computational design to extend protein promiscuity. In: *Biophys Chem* 147 (2010), Mar, Nr. 1-2, S. 13–19

[82] Sun, X. H. ; Baltimore, D.: An inhibitory domain of E12 transcription factor prevents DNA binding in E12 homodimers but not in E12 heterodimers. In: *Cell* 64 (1991), Jan, Nr. 2, S. 459–470

[83] Suzek, B. E. ; Huang, H. ; McGarvey, P. ; Mazumder, R. ; Wu, C. H.: UniRef: comprehensive and non-redundant UniProt reference clusters. In: *Bioinformatics* 23 (2007), Mai, Nr. 10, S. 1282–1288

[84] Tabscott, S. J.: The circuitry of a master switch: Myod and the regulation of skeletal muscle gene transcription. In: *Development* 132 (2005), Jun, Nr. 12, S. 2685–95

[85] Tashiro, M. ; Tejero, R. ; Zimmerman, D. E. ; Celda, B. ; Nilsson, B. ; Montelione, G. T.: High-resolution solution NMR structure of the Z domain of staphylococcal protein A. In: *Journal of Molecular Biology* 272 (1997), Oktober, Nr. 4, S. 573–590

[86] The Apache Software Foundation: *Apache Commons Math.* Online. – URL http://commons.apache.org/math/

[87] Torigoe, H. ; Shimada, I. ; Saito, A. ; Sato, M. ; Arata, Y.: Sequential proton NMR assignments and secondary structure of the B domain of staphylococcal protein A: structural changes between the free B domain in solution and the Fc-bound B domain in crystal. In: *Biochemistry* 29 (1990), Nr. 37, S. 8787–8793

[88] Uhlén, M. ; Guss, B. ; Nilsson, B. ; Gatenbeck, S. ; Philipson, L. ; Lindberg, M.: Complete sequence of the staphylococcal gene encoding protein A. A gene evolved through multiple duplications. In: *J Biol Chem* 259 (1984), Feb, Nr. 3, S. 1695–1702

[89] Vidal, M. A. ; Conde, F. P.: Alternative mechanism of protein A-immunoglobulin interaction the VH-associated reactivity of a monoclonal human IgM. In: *J Immunol* 135 (1985), Aug, Nr. 2, S. 1232–1238

[90] Wagner, Tobias ; Trautmann, Heike ; Martí, Luis: A taxonomy of online stopping criteria for multi-objective evolutionary algorithms. In: *Proceedings of the 6th international conference on Evolutionary multi-criterion optimization*, Springer-Verlag, 2011 (EMO'11), S. 16–30. – ISBN 978-3-642-19892-2

[91] Weintraub, H. ; Dwarki, V. J. ; Verma, I. ; Davis, R. ; Hollenberg, S. ; Snider, L. ; Lassar, A. ; Tapscott, S. J.: Muscle-specific transcriptional activation by MyoD. In: *Genes Dev* 5 (1991), Aug, Nr. 8, S. 1377–1386

[92] Winkler, J. ; Armano, G. ; Dybowski, J. N. ; Kuhn, O. ; Ledda, F. ; Heider, D.: Computational Design of a DNA- and Fc-Binding Fusion Protein. In: *Adv Bioinformatics* 2011 (2011), S. 457578

[93] Yeung, N. ; Lin, Y.-W. ; Gao, Y.-G. ; Zhao, X. ; Russell, B. S. ; Lei, L. ; Miner, K. D. ; Robinson, H. ; Lu, Y.: Rational design of a structural and functional nitric oxide reductase. In: *Nature* 462 (2009), Dec, Nr. 7276, S. 1079–1082

[94] Yin, S. ; Ding, F. ; Dokholyan, N. V.: Eris: an automated estimator of protein stability. In: *Nat Methods* 4 (2007), Nr. 6, S. 466–467

[95] Yin, S. ; Ding, F. ; Dokholyan, N. V.: Modeling backbone flexibility improves protein stability estimation. In: *Structure* 15 (2007), Nr. 12, S. 1567–76

[96] Zitzler, E. ; Brockhoff, D. ; Thiele, L.: The Hypervolume Indicator Revisited: On the Design of Pareto-compliant Indicators Via Weighted Integration. In: *Evolutionary Multi-Criterion Optimization* Bd. 4403, Springer Berlin Heidelberg, 2007, S. 862–876. – ISBN 978-3-540-70927-5

[97] Zitzler, E. ; Thiele, L. ; Laumanns, M. ; Fonseca, C. M. ; Fonseca, V. da: Performance assessment of multiobjective optimizers: an analysis and review. In: *IEEE Transactions on Evolutionary Computation* 7 (2003), April, Nr. 2, S. 117–132

Liste der Publikationen

Artikel

Winkler J, Armano G, Dybowski JN, Kuhn O, Ledda F, Heider D: Computational Design of a DNA- and Fc-Binding Fusion Protein. *Adv Bioinformatics* 2011:457578, 2011.

Winkler J, Hauke S, Pyka M, Heider D: JACVANN: A Java Framework For Complex Valued Artificial Neural Networks. *SIWN 2010* 10:48-55, 2010.

Heider D, Appelmann J, Bayro T, Dreckmann W, Held A, **Winkler J**, Barnekow A, Borschbach M: A Computational approach for the identification of small GTPases based on preprocessed amino acid sequences. *Technol Cancer Res Treat* 8(5):333-41, 2009.

Poster

Winkler J, Heider D: ProteinReactor: A novel algorithm for designing chimeric proteins. *Proc. 25th German Conference on Bioinformatics* Braunschweig, Germany, 2010.

Winkler J, Heider D: ProteinReactor: A Genetic Algorithm to Generate Protein Sequences. *6th NanoBio-Europe* Münster, Germany, 2010.

Danksagung

> *«I want to thank everybody and anybody who ever had anything at all to do with the making of this picture.»*
>
> Morgan Freeman

Auch wenn es hier nicht um einen Oscar geht, möchte ich mich bei allen Menschen bedanken, die in diese Arbeit involviert waren. Mein besonderer Dank gilt:

Meinem Mentor Dominik Heider, der mir alles über Wissenschaft beigebracht hat. Der mich zu dieser Arbeit motiviert und sie gar erst möglich gemacht hat.

Meinem Doktorvater Daniel Hoffmann, für die zündene Idee dieser Arbeit und ein warmes Büro in einer tollen Arbeitsgruppe.

Meinen Kollegen für die Diskussionen, die schöne Zeit und die unzähligen Tassen Kaffee, die uns morgens zusammen auf Trab brachten. Vor allem Olli, für die Hilfe mit AMBER, Niko für die Hilfe mit BrownDye und GROMACS und Christoph, für sein umfassendes Wissen über Epitopsy.

Meinem Admin Manuel, der Unmengen Zeit investiert hat, die Computersysteme am Laufen zu halten, um auch mal in Zeitnot Simulationen durchführen zu können.

Meinen Eltern Barbara und Horst, die nie an mir zweifelten und mich mein ganzes Leben immer unterstützt haben.

Meiner Liebe Anika, für die Geduld, die Hilfe und den Ansporn. Vor allem aber dafür, dass es dich gibt.

Curriculum Vitæ

Der Lebenslauf ist in der Online-Version aus Gründen des Datenschutzes nicht enthalten.

Curriculum Vitæ (Seite 2)

Der Lebenslauf ist in der Online-Version aus Gründen des Datenschutzes nicht enthalten.

Erklärungen

Erklärung:
Hiermit erkläre ich gemäß § 6 Abs. (2) f) der der Promotionsordnung der Fakultäten für Biologie und Geografie, Chemie und Mathematik zur Erlangung des Dr. rer. nat., dass ich das Arbeitgebiet, dem das Thema "Sequenzoptimierung eines synthetischen bifunktionalen Proteins durch multikriterielle genetische Algorithmen" zuzuordnen ist, in Forschung und Lehre vertrete und den Antrag von Jonas Winkler befürworte und die Betreuung auch im Falle eines Weggangs, wenn nicht wichtige Gründe dem entgegenstehen, weiterführen werde.

Essen, den _____ _____

Unterschrift eines Mitgliedes der Universität Duisburg-Essen

Erklärung:
Hiermit erkläre ich, gem. § 7 Abs. (2) c) + e) der Promotionsordnung Fakultäten für Biologie und Geografie, Chemie und Mathematik zur Erlangung des Dr. rer. nat., dass ich die vorliegende Dissertation selbständig verfasst und mich keiner anderen als der angegebenen Hilfsmittel bedient habe.

Essen, den _____ _____

Unterschrift des Doktoranden

Erklärung:
Hiermit erkläre ich, gem. § 7 Abs. (2) d) + f) der Promotionsordnung der Fakultäten für Biologie und Geografie, Chemie und Mathematik zur Erlangung des Dr. rer. nat., dass ich keine anderen Promotionen bzw. Promotionsversuche in der Vergangenheit durchgeführt habe und dass diese Arbeit von keiner anderen Fakultät/Fachbereich abgelehnt worden ist.

Essen, den _____ _____

Unterschrift des Doktoranden